Полное руководство по Burp Suite

Узнайте, как обнаруживать уязвимости в приложениях

Sagar Rahalkar

Apress®

Полное руководство по Burp Suite

Узнайте, как обнаруживать уязвимости в приложениях

Sagar Rahalkar

Apress®

Содержание

Об авторе

Sagar Rahalkar - опытный специалист по информационной безопасности с более чем 13-летним опытом работы в различных вертикалях информационной безопасности. Его специализация в основном связана с AppsSec, расследованиями киберпреступности, оценкой уязвимостей, тестированием на проникновение и IT-GRC. Он получил степень магистра в области компьютерных наук и несколько признанных в отрасли сертификатов, таких как CISM, ISO 27001LA и ECSA. Он был тесно связан с индийскими правоохранительными органами в течение более трех лет, занимаясь расследованиями цифровых преступлений и соответствующей подготовкой, и он получил награды от высокопоставленных должностных лиц полиции и оборонных организаций в Индии. Он также является автором и рецензентом нескольких публикаций.

О техническом рецензенте

Parag Patil (www.linkedin.com/in/paragpatil2006) является специалистом по информационной безопасности, в настоящее время работающим с Coupa, управляющим безопасностью для платформы SaaS.

В течение более чем последних 12 лет Parag активно работала над цифровой криминалистикой, IAM, мониторингом безопасности/SecOps, тренингами по безопасности, аудитами соответствия требованиям безопасности, управлением уязвимостями тестированием на проникновение, исследованиями информационной безопасности и ISMS/управлением. Он является автором тестов CIS для AWS, Azure и GCP.

Благодарности рецензента: Спасибо моим наставникам (Dattatray Bhat, Yogesh Patil, Steve O'Callaghan, Shailesh Athlye, и Hans Gustavson), которые верили в меня и предоставили все возможные возможности для обучения и профессионального роста в области информационной безопасности.

Спасибо Mahesh Navaghane, Sagar Rahalkar (автору этой книги), Aditi Sahasrabuddhe (моей сестре), Monika (моей жене) и Ira (моей дочери) за их приверженность к тому, чтобы сделать меня, возможно, самым счастливым человеком, которого я когда-либо знал.

Введение

Число приложений растет, как и число уязвимостей приложений. Предприятия уделили большое внимание обеспечению безопасности приложений. Хотя существует множество решений и продуктов для обеспечения безопасности приложений, Burp Suite действительно является инструментом выбора для многих.

Burp Suite - это простой, но мощный инструмент, используемый для тестирования безопасности приложений. Он широко используется для ручного тестирования безопасности не только веб-приложений, но и API и мобильных приложений. Для эффективного тестирования безопасности веб-приложений необходимо понимать различные уязвимости веб-приложений; в то же время необходимо также иметь глубокое понимание инструментов, используемых для тестирования. Эта книга поможет вам всесторонне понять Burp Suite, чтобы его можно было использовать именно для выявления уязвимостей.

Книга начинается с основ Burp Suite и поможет вам настроить среду тестирования. В следующих главах рассматриваются основные структурные блоки Burp Suite и подробно рассматриваются его различные компоненты, такие как intruder, repeater, decoder, comparer, sequencer и т. д. В последних главах мы рассмотрим другие полезные функции, такие как infiltrator, collaborator, scanner, extender, и использование Burp Suite для тестирования безопасности API и мобильных приложений.

Глава 1

Введение в Burp Suite

Безопасность приложений в значительной степени изменилась за последнее десятилетие или около того. Десять лет назад найти SQL-инъекции в приложениях было проще, чем сегодня. Приложения были более подвержены уязвимостям, поскольку было меньше средств защиты и меньше осведомленности среди разработчиков. Однако сегодня ситуация кардинально изменилась. Разработчики гораздо лучше знают и осознают безопасности, и средства контроля безопасности применяются на протяжении всего жизненного цикла разработки программного обеспечения (SDLC), что делает конечное приложение сравнительно безопасным.

Хотя процессы разработки стали более безопасными, современные приложения не ограничиваются только веб-сайтами. Современные приложения имеют открытые сервисы и интерфейсы прикладного программирования (API), а также мобильное и облачное использование. Это явно увеличивает сложность и площадь атаки.

Для тестировщика безопасности приложений, жизненно важно найти все возможные уязвимости во всей экосистеме приложений.

Некоторые основы безопасности веб-приложений

Подробное описание безопасности приложений и различных уязвимостей выходит за рамки этой книги. В этой книге мы специально сосредоточимся на том, как наиболее эффективно использовать инструмент Burp Suite.

S. Rahalkar, *A Complete Guide to Burp Suite*,
https://doi.org/10.1007/978-1-4842-6402-7_1

Тем не менее, мы быстро рассмотрим, в чем заключаются общие и основные уязвимости приложений. Стандартом де-факто, на который ссылаются для уязвимостей приложений, является OWASP. OWASP расшифровывается как "Проект безопасности открытых веб-приложений". Последний список Топ-10 уязвимостей веб-приложений был опубликован в 2017 году. Уязвимости заключаются в следующем:

1. **Инъекция** – Это включает в себя уязвимости, которые используются путем отправки ненадежных входных данных интерпретатору либо как часть запроса, либо как часть команды. Специально разработанные приемы ввода это то, что интерпретатор использует при выполнении команд или даже при предоставлении несанкционированного доступа к данным. Наиболее распространенным типом инъекции является инъекция базы данных. Другие типы включают инъекцию команд операционной системы (ОС) или инъекцию LDAP и т. д.
2. **Нарушение аутентификации** – Это включает уязвимости, возникающие из-за плохой реализации функций аутентификации и управления сеансами. Использование таких уязвимостей может предоставить злоумышленникам доступ к паролям, учетным данным, токенам сеанса, ключам и т.д.
3. **Раскрытие конфиденциальных данных** – Во многих случаях приложениям не хватает средств контроля для защиты конфиденциальных пользовательских данных, таких как личная информация (PII), данные о состоянии здоровья или даже финансовых данных. Злоумышленники могут украсть такие конфиденциальные данные. Отсутствие шифрования данных в состоянии покоя и при передаче вызывает большую часть уязвимостей, связанных с раскрытием конфиденциальных данных.
4. **Внешние сущности XML** – Это особый тип уязвимости, при котором злоумышленник использует тег сущности в XML-документах для запуска нескольких атак, таких как раскрытие конфиденциальных внутренних файлов, отказ в обслуживании, удаленное выполнение кода и т.д.

5. **Нарушение контроля доступа** – Даже если пользователь прошел проверку подлинности с действительными учетными данными, может не потребоваться доступ ко всему приложению. Авторизация определяет, к чему может получить доступ аутентифицированный пользователь. Нарушенная авторизация предоставляет злоумышленнику несанкционированный доступ для просмотра других учетных записей пользователей, конфиденциальных файлов или даже изменения данных других пользователей.

6. **Неправильная конфигурация безопасности** – проблемы с неправильной настройкой безопасности наиболее распространены в базовой инфраструктуре, такой как веб-серверы. Небезопасные конфигурации, учетные данные по умолчанию, файлы резервных копий, нежелательные службы, открытое облачное хранилище, отсутствующие заголовки безопасности и флаги файлов cookie, а также отсутствующие исправления безопасности - все это относится к категории неправильной конфигурации безопасности.

7. **Межсайтовый скриптинг** - это действительно классическая уязвимость веб – приложений, которая так долго была частью списка OWASP. Обычно это происходит, когда злоумышленник может внедрить и выполнить сценарий через поле ввода приложения. Эта атака может быть использована для захвата пользовательских сеансов путем кражи файлов cookie, порчи веб-сайтов и т.д. Распространенные типы межсайтовых сценариев включают хранимые, отраженные и основанные на DOM.

8. **Небезопасная десериализация** – Злоумышленники могут манипулировать процессом сериализации и десериализации объектов для внедрения вредоносных полезных нагрузок, приводящих к выполнению кода.

9. **Использование компонентов с известными уязвимостями** – Разработчики очень часто импортируют и используют сторонний код, чтобы избежать изобретения велосипеда. Однако иногда сторонний код сопровождается присущими ему уязвимостями. Примером может служить использование библиотеки OpenSSL, которая уязвима для Heart Bleed атак.
10. **Недостаточное ведение журнала и мониторинг** – Довольно часто приложениям не хватает возможностей для регистрации событий, которые помогли бы в случае инцидента. В отсутствии ведения журнала аудита и возможностей обнаружения злоумышленники могут просто продолжить проникновение, не будучи обнаруженными или не вызывая тревоги.

Хотя список OWASP Топ-10, вероятно, является первым местом, куда следует обратиться за уязвимостями веб-приложений, за пределами этого списка Топ-10 существует множество потенциальных уязвимостей. Ниже приведены некоторые из настоятельно рекомендуемых ссылок, чтобы получить более широкую картину для тестирования безопасности приложений:

1. **OWASP Testing Guide** – Это руководство является очень обширным ресурсом, охватывающим множество примеров тестирования безопасности, и очень полезным справочным руководством. Он доступен по адресу https://owasp.org/www-project-web-security-testing-guide/assets/archive/OWASP_Testing_Guide_v4.pdf
2. **SANS Top 25 Programming Errors** – Помимо списка OWASP Top 10, SANS опубликовал список из 25 самых опасных ошибок программирования. Он доступен по адресу https://www.sans.org/top25-software-errors

3. **OWASP API Top 10** - Интерфейсы прикладного программирования (API) очень часто используются в наши дни и имеют некоторые уникальные уязвимости. OWASP опубликовал специальный список уязвимостей Топ-10 API, который доступен по адресу https://owasp.org/www-project-api-security/
4. **OWASP Mobile Top 10** - Мобильные приложения имеют различные типы уязвимостей, а некоторые даже различаются в зависимости от типа платформы. Однако наиболее распространенные и основные уязвимости мобильных устройств доступны по адресу https://owasp.org/www-project-mobile-top-10/
5. **OWASP IoT Top 10** – Сегодня даже бытовые устройства становятся умнее и подключаются. Такие устройства Интернета вещей (IoT) подвержены множеству уязвимостей. OWASP опубликовал список 10 лучших уязвимостей IoT, доступный по адресу https://owasp.org/www-project-internet-of-things/

Краткое введение в Burp Suite

Появление Burp Suite датируется 2004 годом, когда Dafydd Stuttard оценил потребность в надежном инструменте тестирования безопасности веб-приложений. За последние 16 лет этот инструмент стремительно развивался и добавил множество возможностей, которые приносят пользу сообществу тестирования безопасности. Burp Suite, несомненно, стал предпочтительным инструментом для тестирования безопасности веб-приложений. Кроме того, он эволюционировал таким образом, что теперь его можно использовать для поиска уязвимостей в API и мобильных приложениях.

Нужен ли Burp Suite

Сегодня рынок инструментов сканирования и тестирования безопасности приложений быстро растет. Существует так много доступных инструментов, как коммерческих, так и бесплатных, от разных поставщиков, поддерживающих различные технологии и функции. Большинство этих инструментов ориентированы на автоматическое сканирование программного обеспечения для поиска уязвимостей. Это достигается либо путем запуска сканера после обхода целевого приложения, либо путем интеграции сканера непосредственно в цикл DevOps. Хотя это, безусловно, является преимуществом и повышает эффективность сканирования при минимальном ручном вмешательстве, существуют определенные уязвимости, которые можно лучше понять и использовать с помощью ручного тестирования.

Ручное тестирование в значительной степени зависит от двух факторов: навыков тестировщика и инструмента, используемого для тестирования. Такой инструмент, как Burp Suite, значительно облегчает выполнение задач ручного тестирования с точки зрения инструментов. Это обеспечивает мощную и гибкую платформу, на которой тестировщик может эффективно находить и использовать потенциальные уязвимости. Таким образом, для сканирования и тестирования безопасности приложений лучшей стратегией было бы использовать комбинацию как автоматического, так и ручного тестирования. Burp Suite обладает отличными возможностями ручного тестирования наряду с автоматизированным сканером. Таким образом, это дает тестировщику преимущества ручного тестирования, а также автоматического сканирования уязвимостей.

Издания

Как и большинство других инструментов, Burp Suite поставляется в различных формах. У разных пользователей могут быть разные потребности, и один формат может подходить не всем. Учитывая различные потребности пользователей, Burp Suite выпускается в трех различных версиях.

1. **Burp Suite Community Edition –** Это самая базовая версия, которую можно бесплатно скачать и использовать. Он поставляется с ограниченным набором инструментов и функций для начала тестирования безопасности веб-приложений. Если вы совершенно новичок в области безопасности приложений и хотите изучить основы, то Burp Suite Community Edition, безусловно, является очень хорошей стартовой точкой. В нем есть хорошие инструменты и функции, необходимые для базового ручного тестирования безопасности веб-приложений, такие как прокси-сервер, перехват, ретранслятор, кодирование и декодирование данных и т.д.
2. **Burp Suite Professional Edition –** Если у вас есть очень хорошее представление о безопасности веб-приложений и вам регулярно требуется тестировать приложения в рамках вашей профессии, то определенно рекомендуется использовать Burp Suite Professional Edition. Профессиональная версия Burp Suite поставляется вместе со многими расширенными функциями, которые значительно улучшают вашу способность находить потенциальные уязвимости в приложениях. Это наиболее подходящее издание для отдельных специалистов, которым требуются отличные возможности ручного и автоматизированного тестирования безопасности. Некоторые из дополнительных функций включают следующее:
    - Тестирование out-of-band уязвимостей
    - Расширенные возможности грубой силы и фаззинга
    - Быстрое создание эксплойтов для CSRF, Clickjacking и т.д.

- Автоматическое сканирование на наличие уязвимостей
- Полезные расширения для дальнейшего расширения возможностей обнаружения уязвимостей

Более подробную информацию о профессиональном издании Burp Suite можно найти здесь - https://portswigger.net/burp/pro

3. **Burp Suite Enterprise Edition** – В то время как Burp Suite Community Edition и Burp Suite Professional Edition были предназначены для отдельных специалистов, Burp Suite Enterprise Edition полезно организациям, которые ищут интеграцию сканирования безопасности в программные конвейеры. В нем нет инструментов ручного тестирования по сравнению с более ранними выпусками. Это издание рекомендуется для предприятий, которые ищут решения DevSecOps.

В рамках этой книги мы рассмотрим Burp Suite Professional Edition.

Альтернативы Burp Suite

Мы уже обсуждали, что рынок инструментов сканирования безопасности приложений в значительной степени растет. Хотя Burp Suite удовлетворяет большинство потребностей в ручном и автоматическом тестировании, он конкурирует с некоторыми другими инструментами, такими как те, которые показаны в таблице 1-1.

Таблица 1-1. Инструменты Сканирования

Коммерческие	Бесплатные
Acunetix	OWASP ZAP
Netsparker	W3af
IBM AppScan	Arachni
WebInspect	Iron Wasp

Более подробную информацию и сравнительный анализ различных инструментов тестирования безопасности приложений можно найти по адресу https://www.gartner.com/reviews/market/application-security-testing

Обзор функций высокого уровня

Burp Suite Professional Edition поставляется с широким спектром функций для ручного тестирования на проникновение, а также для автоматического сканирования. Некоторые из полезных функций включают следующее:

1. **Ручное тестирование на проникновение** – Перехват и подделка запросов (HTTP / HTTPS), ручное тестирование на наличие out-of-band уязвимостей, тестирование веб-сокетов, проверка надежности токенов, простое тестирование уязвимостей clickjacking и подделки межсайтовых запросов (CSRF).
2. **Расширенные автоматизированные атаки** – Пассивное и активное сканирование для поиска потенциальных уязвимостей, расширенные возможности для применения грубой силы и фаззинга.
3. **Продуктивность** – Подробный анализ сообщений, эффективные параметры проекта, инструменты, позволяющие сделать код более читаемым, легкие и простые отчеты об уязвимостях.

4. **Расширения** – Хранилище приложений Burp Suite для установки расширений для значительного расширения существующих возможностей инструмента.

Мы рассмотрим вышеперечисленные функции более подробно по мере продвижения по книге.

Резюме

Мы начали эту главу с объяснения того, как безопасность приложений развивалась за последнее десятилетие или около того. Затем мы рассмотрели некоторые из основных уязвимостей веб-приложений. Затем мы попытались понять необходимость такого инструмента, как Burp Suite, а затем его выпуски и альтернативы. Наконец, мы завершили обзор функций высокого уровня, предоставляемых Burp Suite Professional.

В следующей главе мы начнем с установки и настройки инструмента.

Упражнения

- Подробно ознакомьтесь с Топ-10 уязвимостей OWASP и руководством по тестированию OWASP (OWASP Testing Guide).
- Подробнее об особенностях всех выпусков Burp Suite читайте на официальном сайте - https://portswigger.net/

Глава 2

Настройка среды

В предыдущей главе мы обсудили некоторые основы безопасности приложений и необходимость использования таких инструментов, как Burp Suite. В этой главе мы начнем с настройки нашей среды в Burp Suite.

Установка Burp Suite

Прежде чем мы попытаемся установить или запустить Burp Suite, нам необходимо убедиться, что в системе установлена Java. Это необходимое предварительное условие для запуска Burp Suite. В системе Windows вы можете просто открыть командную строку и ввести команду “java –version”, чтобы проверить, установлена ли Java, как показано на рис.2-1.

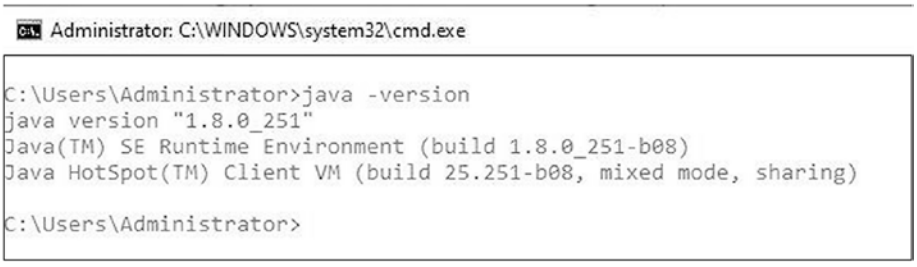

Рис.2-1. *Проверьте, установлена ли Java*

S. Rahalkar, *A Complete Guide to Burp Suite*,
https://doi.org/10.1007/978-1-4842-6402-7_2

Если в вашей системе не установлена Java, вы можете загрузить и установить Java с https://www.oracle.com/java/technologies/javase-jre8-downloads.html

Как только мы убедимся, что Java установлена в нашей системе, мы сможем приступить к работе с Burp Suite. Сначала нам нужно загрузить Burp Suite с https://portswigger.net/burp/releases/community/latest как показано на рис. 2-2.

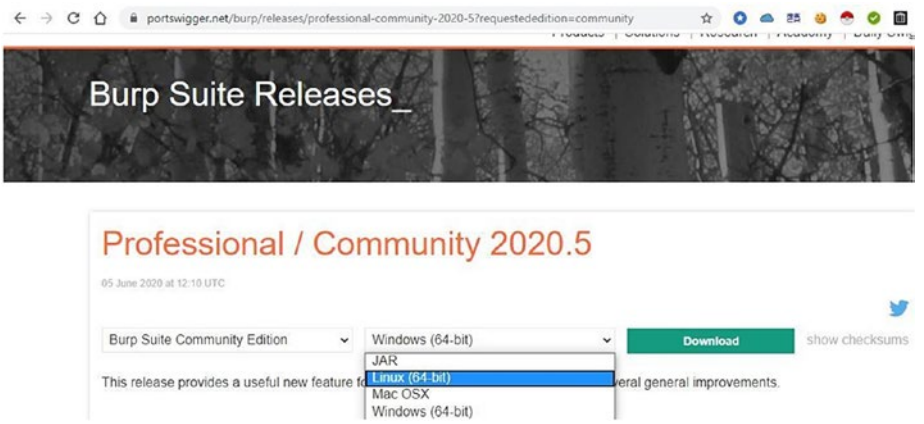

Рис.2-2. *Типы загрузок Burp Suite*

Вы заметите, что существует несколько форм, в которых вы можете загрузить Burp Suite. Существуют отдельные установщики для Linux, Mac OSX и Windows. Также есть возможность загрузить файл JAR, который можно использовать непосредственно для запуска Burp Suite без установки. Загрузка файла JAR - это самый простой способ начать работу. Если вы решите загрузить установщик, он будет таким же, как и любой другой установщик программного обеспечения, и установит Burp Suite в несколько кликов. Однако в обоих случаях требуется установить Java. Как только файл JAR будет загружен, вы можете просто дважды щелкнуть по нему, чтобы запустить Burp Suite.

Иногда во время выполнения крупных проектов может случиться так, что в Burp Suite закончится память. Чтобы решить эту проблему, можно запустить Burp Suite, выделив фиксированный объем памяти при запуске. Это гарантирует, что после запуска у него не закончится память. Это можно сделать с помощью команды "java -jar -Xmx2G /path/to/burp.jar" где 2G указывает на 2 ГБ памяти. Этот шаг совершенно необязателен. Мы можем пропустить его и напрямую запустить файл JAR, чтобы запустить Burp Suite с конфигурацией по умолчанию.

Если все необходимые условия выполнены правильно, мы получим экран запуска, как показано на рис.2-3.

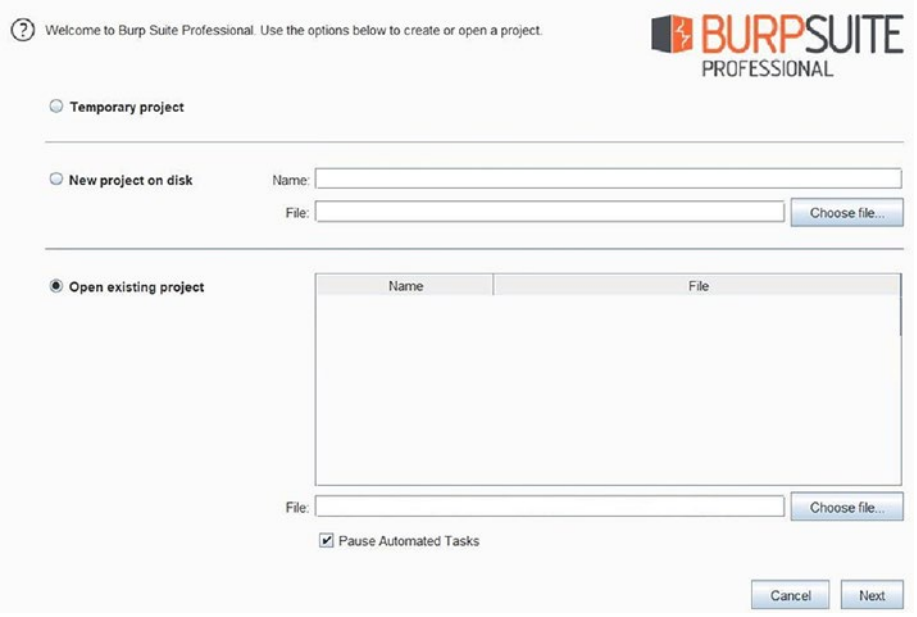

Рис.2-3. *Экран запуска Burp Suite*

Настройка уязвимого целевого веб-приложения

Пока мы настраиваем Burp Suite в нашей системе, важно иметь целевое приложение, в котором вы будете использовать этот инструмент. Если вы являетесь профессионалом, работающим над тестированием безопасности приложений и тестированием на проникновение, вам будет разрешено использовать Burp Suite в тестируемом приложении. Однако, если вы всего лишь новичок, пытающийся начать изучение Burp Suite, вам понадобится какое-то целевое приложение, в котором вы могли бы проверить свои навыки. Помните, что запуск Burp Suite в приложении, в котором вы не имеете права, может привести к юридическим проблемам. Поэтому, с точки зрения обучения, важно опробовать свои навыки Burp Suite только в тестовом приложении. Существует несколько доступных альтернатив, как показано ниже:

1. **Локальный запуск OWASP Juice Shop –** OWASP Juice Shop - Это современное веб-приложение, которое намеренно сделано уязвимым. Это может быть отличной отправной точкой. Самый простой способ запустить OWASP Juice Shop - это использовать его образ docker. Образ docker доступен по адресу https://hub.docker.com/r/bkimminich/juice-shop. Вы можете просто вытащить образ и запустить его в docker engine на любой платформе (Windows / Linux /macOS).
2. **Онлайн версия OWASP Juice Shop –** Как новичку, всегда рекомендуется создать свою собственную копию Juice Shop; однако, если вы хотите быстро опробовать ее перед настройкой, вы можете попробовать онлайн-версию по адресу https://juice-shop.herokuapp.com/#/

3. **Damn Vulnerable Web Application (DVWA)** – Еще одно замечательное приложение, которое было намеренно сделано уязвимым для тестирования, - это DVWA. Вы можете быстро настроить DVWA с помощью docker или на локальном веб-сервере. Подробные инструкции по настройке и использованию DVWA доступны по адресу https://github.com/ethicalhack3r/DVWA
4. **Damn Vulnerable Web Services** – OWASP Juice Shop и DVWA будет достаточно для ваших потребностей в изучении уязвимостей веб-приложений. Однако, если вы хотите более конкретно изучить уязвимости в веб-службах, то Damn Vulnerable Web Services - хороший вариант. Более подробную информацию о настройке и использовании можно найти по адресу https://github.com/snoopysecurity/dvws

Настройка браузера

Теперь, когда мы запустили Burp Suite, нам нужно настроить наш браузер так, чтобы он работал вместе с ним. Сначала давайте рассмотрим обычный сценарий без Burp Suite на картинке, как показано на рис.2-4.

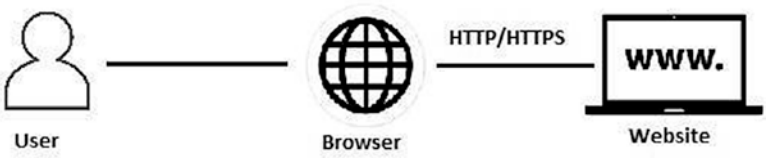

Рис.2-4. *Пользователь, заходящий на веб-сайт напрямую без Burp Suite*

Ссылаясь на изображение выше, на очень высоком уровне и простыми словами, происходит следующая последовательность событий:

1. Конечный пользователь открывает любой браузер по своему выбору.
2. Затем пользователь вводит URL-адрес веб-сайта, который он/она хочет просмотреть.
3. Браузер обрабатывает URL-адрес веб-сайта и отображает веб-сайт для пользователя (серия запросов и ответов происходит в фоновом режиме).

Теперь давайте рассмотрим другой сценарий, в котором мы настроили Burp Suite с браузером, как показано на рис.2-5.

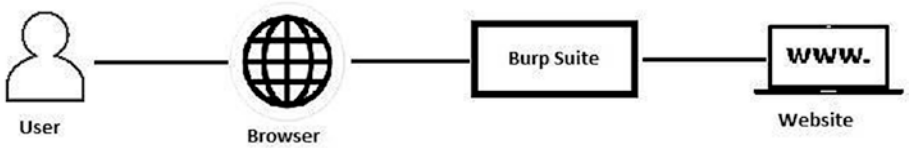

Рис.2-5. *Пользователь, заходящий на веб-сайт с помощью Burp Suite*

Ссылаясь на изображение выше, на очень высоком уровне и простыми словами, происходит следующая последовательность событий:

1. Конечный пользователь открывает любой браузер по своему выбору.
2. Затем пользователь вводит URL-адрес веб-сайта, который он/она хочет просмотреть.
3. Браузер перенаправляет запрос в Burp Suite, который затем перенаправляет запрос на целевой веб-сайт.
4. Целевой веб-сайт отвечает на запрос и отправляет ответ обратно в Burp Suite, который затем передает ответ для отображения в браузере.

Таким образом, в этом сценарии Burp Suite действует как "Человек посередине" между браузером и целевым веб-сайтом. Burp Suite способен перехватывать и блокировать весь трафик, проходящий через него.

Теперь мы посмотрим, как мы можем настроить самые популярные браузеры для работы с Burp Suite.

Firefox

Для настройки Firefox с помощью Burp Suite:

Перейдите в раздел Инструменты и параметры, как показано на рис.2-6.

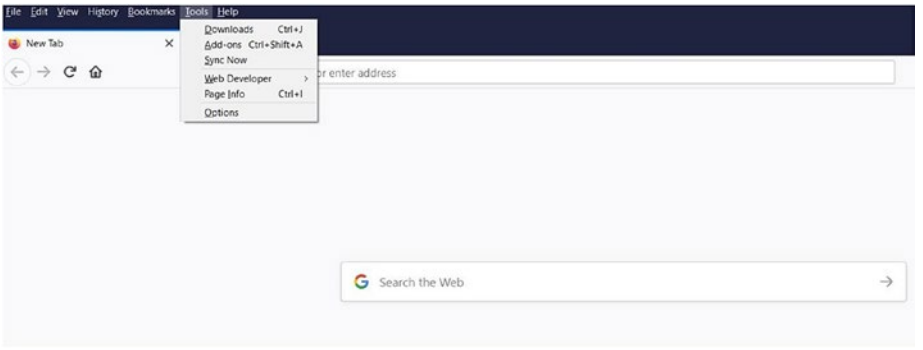

Рис.2-6. *Навигация по меню "Инструменты" и "Параметры" в Firefox*

В поле поиска введите ключевое слово "сеть", как показано на рисунке 2-7, и нажмите "Настройки".

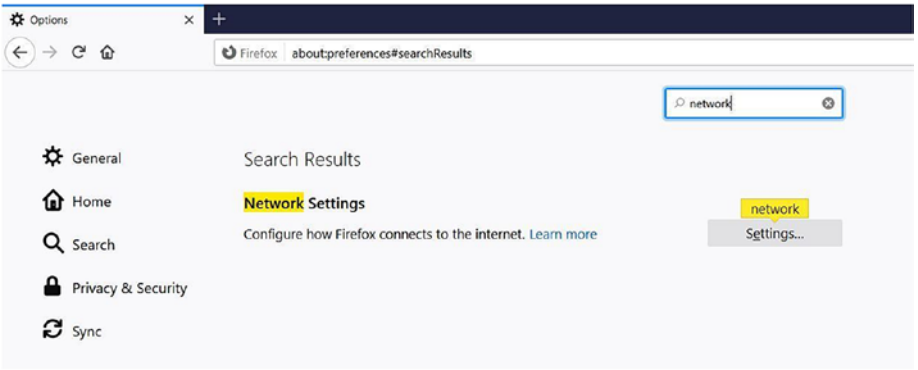

Рис.2-7. *Поиск "Сетевых настроек" в настройках Firefox*

Выберите "Настройка прокси вручную", как показано на рисунке 2-8, и введите IP-адрес как 127.0.0.1 (или localhost), а порт как 8080.

Примечание: По умолчанию прокси-сервер Burp Suite прослушивает порт 8080. Это можно настроить, и мы увидим это в следующей главе. Однако один и тот же номер порта необходимо ввести как в браузере, так и в Burp Suite, если вы хотите изменить его.

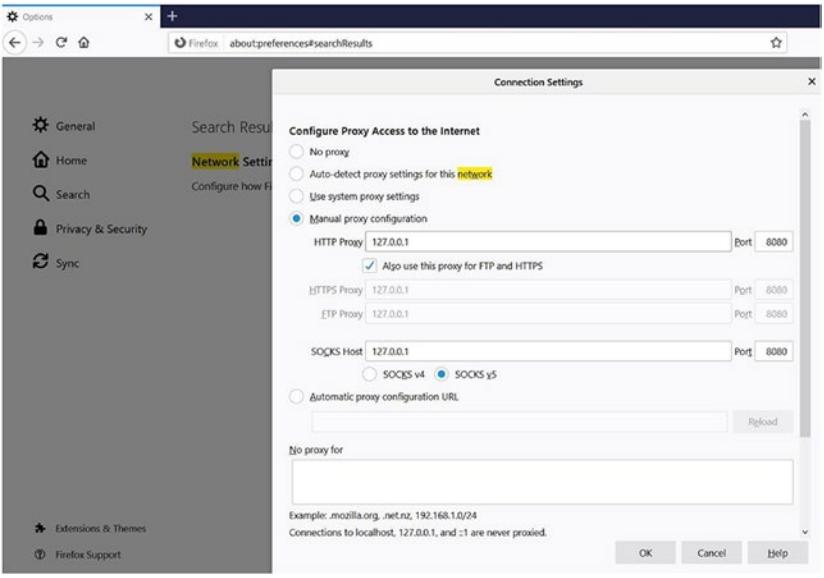

Рис.2-8. *Настройка конфигурации прокси-сервера вручную*

Просто нажмите "ОК", как только будут установлены данные прокси-сервера.

Chrome

Для настройки Chrome с помощью Burp Suite:

Нажмите на три вертикальные точки в правом углу и выберите "Настройки", как показано на рис.2-9.

Рис.2-9. *Переход к настройкам Chrome*

Найдите ключевое слово "прокси", как показано на рис. 2-10, и нажмите на опцию "Открыть настройки прокси-сервера вашего компьютера".

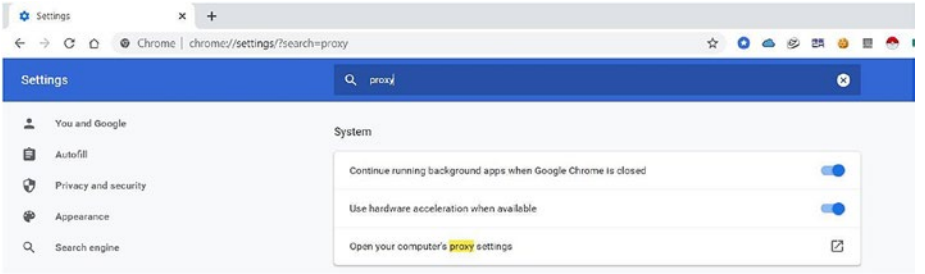

Рис.2-10. *Открытие настроек прокси-сервера в Chrome*

Теперь включите опцию "Использовать прокси-сервер" и введите адрес и номер порта, как показано на рис.2-11.

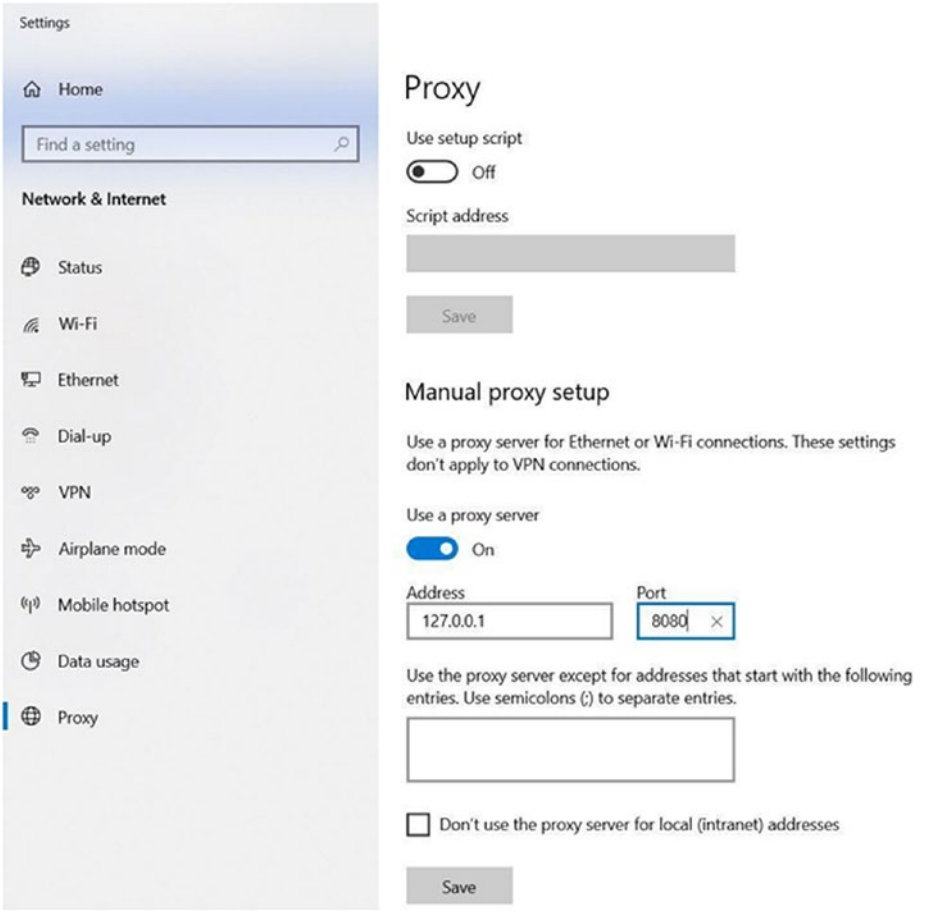

Рис.2-11. *Настройка прокси-сервера системы*

Как только прокси-сервер настроен, просто нажмите на опцию "Сохранить".

Edge

Для настройки Edge с помощью Burp Suite:

Нажмите на три горизонтальные точки в правом углу и выберите "Открыть настройки прокси", как показано на рис.2-12.

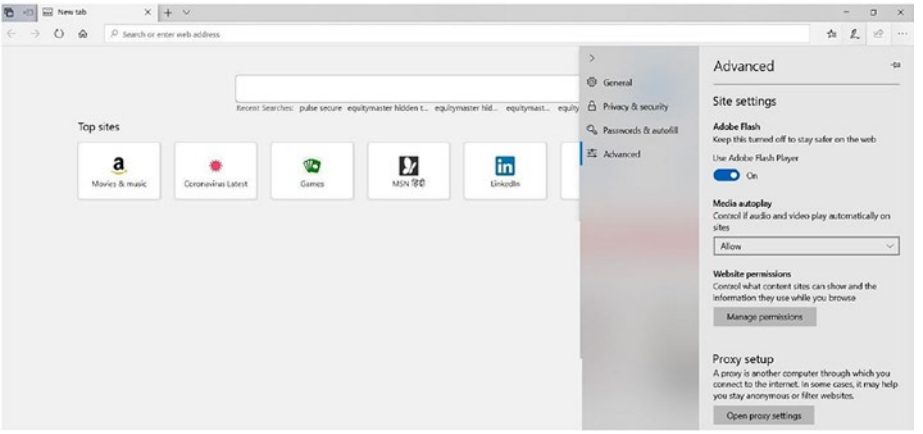

Рис.2-12. *Открытие настроек прокси-сервера в браузере Edge*

Теперь включите опцию "Использовать прокси-сервер" и введите адрес и номер порта, как показано на рисунке 2-13.

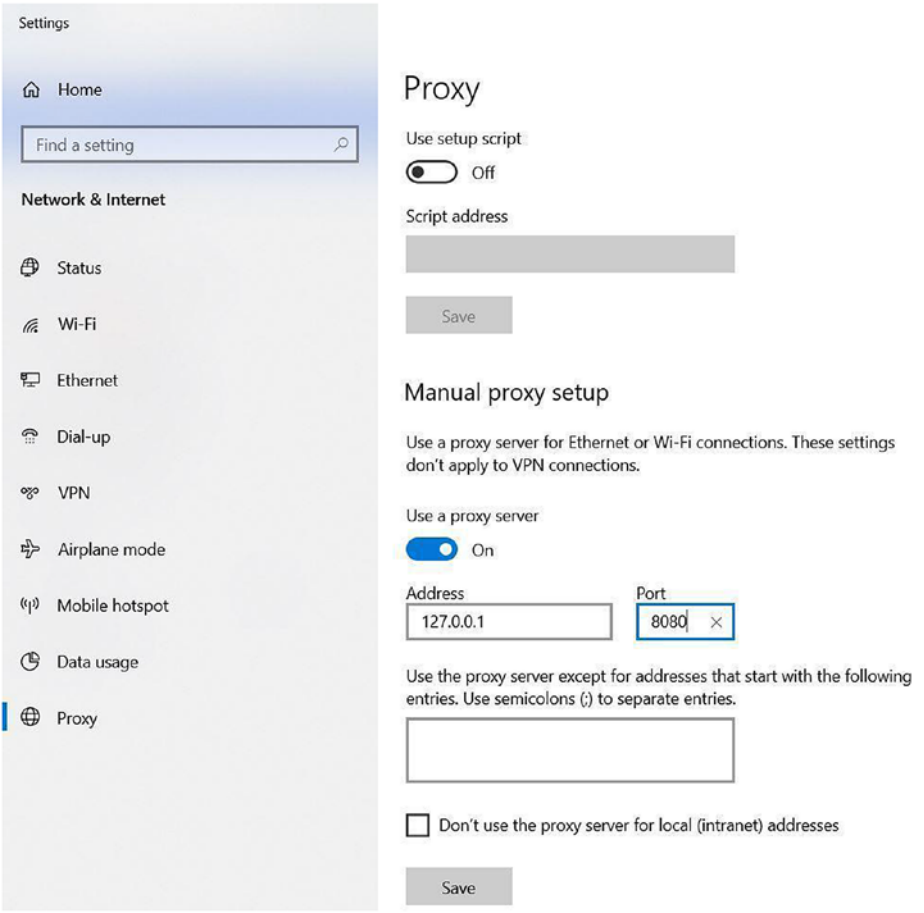

Рис.2-13. *Настройка прокси-сервера системы*

Как только прокси-сервер настроен, просто нажмите на опцию "Сохранить".

Opera

Для настройки Opera с помощью Burp Suite:

Нажмите на настройки в правом верхнем углу и выберите опцию "Перейти в настройки браузера", как показано на рис.2-14.

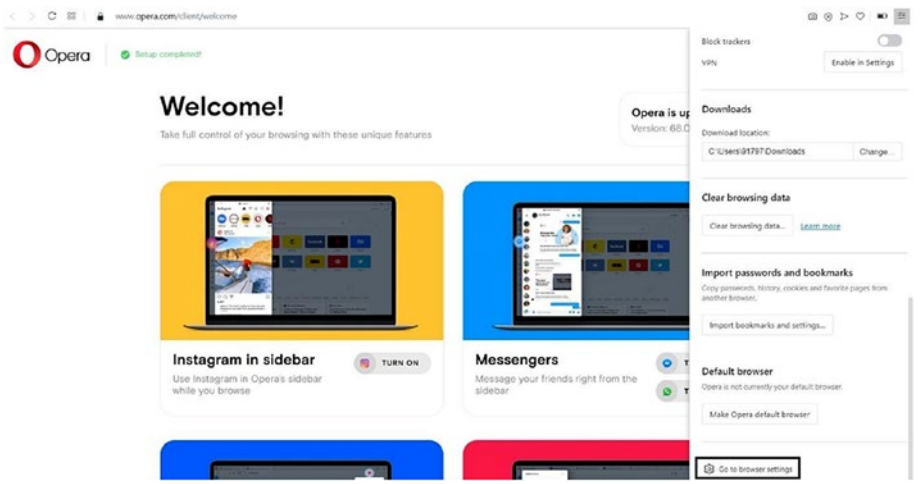

Рис.2-14. *Открытие настроек браузера в Opera*

В поле поиска введите прокси-сервер, а затем выберите опцию "Открыть настройки прокси-сервера вашего компьютера", как показано на рис. 2-15.

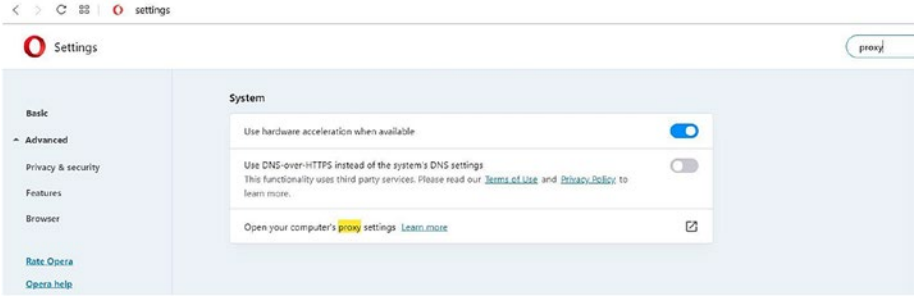

Рис.2-15. *Открытие системных настроек прокси-сервера*

Теперь включите опцию "Использовать прокси-сервер" и введите адрес и номер порта, как показано на рис.2-16.

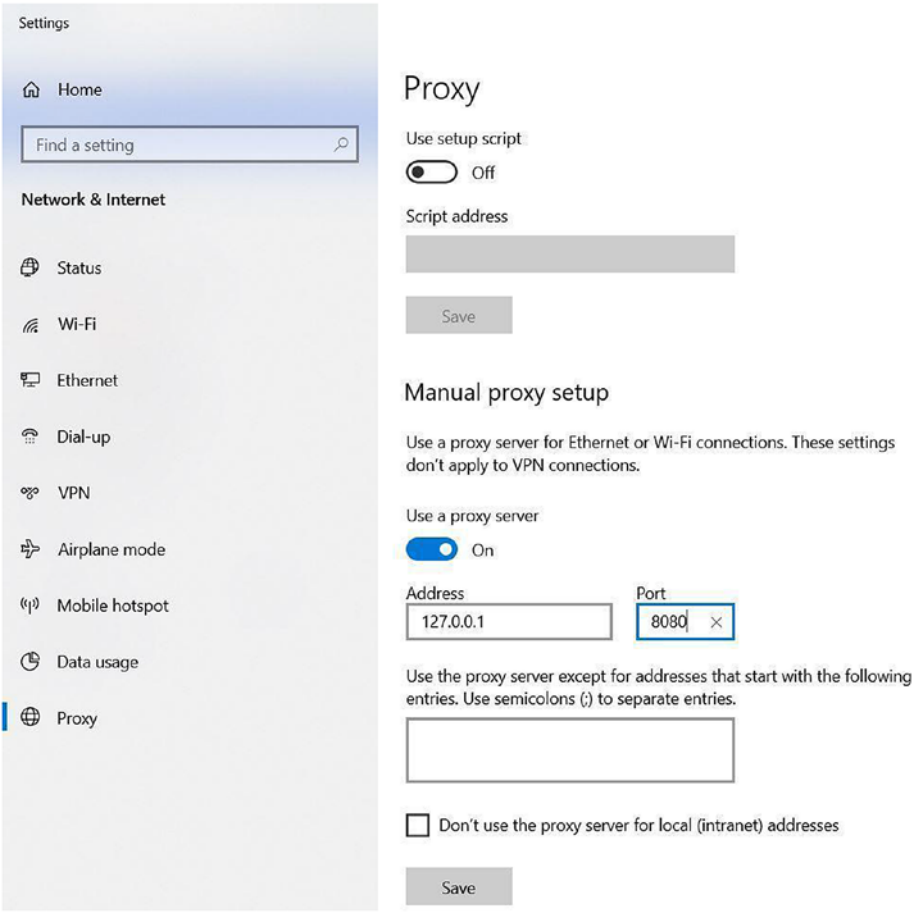

Рис.2-16. *Настройка системного прокси-сервера*

Как только прокси-сервер настроен, просто нажмите на опцию "Сохранить".

Теперь мы видели, как настроить такие браузеры, как Firefox, Chrome, Edge и Opera, для работы вместе с Burp Suite. Для этого просто требуется настроить параметр сетевого прокси-сервера. Однако важно отметить, что после настройки прокси-сервера браузера весь трафик, исходящий из браузера, в обязательном порядке будет проходить через Burp Suite. Если вы работаете на нескольких вкладках в браузере

и тестируете приложение на одной вкладке, получая доступ к электронной почте на другой, весь этот трафик будет перенаправляться через Burp Suite. В случае, если вы хотите пропускать только выборочный трафик через Burp Suite, вам необходимо использовать дополнительные плагины для браузера, такие как те, которые показаны в таблице 2-1.

Таблица 2-1. *Дополнительные плагины браузера для прокси*

Firefox	Proxy SwitchyOmega, FoxyProxy
Chrome	Proxy SwitchyOmega, FoxyProxy
Edge	N/A
Opera	Proxy Switcher & Manager

Вышеперечисленные плагины просты в использовании и позволяют выборочному трафику проходить через Burp Suite. Использование этих плагинов совершенно необязательно.
Если вы не хотите использовать эти плагины, вы можете просто использовать два отдельных экземпляра браузера, один для тестирования приложений, а другой для личного использования. Или также можно выделить только необходимый трафик в Burp Suite, который мы изучим в следующей главе.

Резюме

В этой главе мы рассмотрели, как загрузить, установить и начать работу с инструментом Burp Suite. Затем мы изучили различные варианты, доступные для настройки уязвимых целей, чтобы попрактиковаться в навыках Burp Suite. Мы также узнали, как настроить различные браузеры для работы вместе с Burp Suite.

В следующей главе мы рассмотрим, как настроить некоторые основные параметры Burp Suite, такие как Прокси-сервер, Параметры пользователя и Параметры проекта.

Упражнения

- Загрузите последнюю версию Burp Suite.
- Попробуйте запустить Burp Suite из командной строки, выделив пользовательский размер памяти.
- Попробуйте и изучите, как использовать плагин FoxyProxy для Firefox и Chrome.

Глава 3

Прокси, параметры пользователя и проекта

В предыдущей главе мы рассмотрели некоторые основы установки Burp Suite, настройки уязвимого приложения и настройки различных браузеров для совместной работы. В этой главе мы начнем с некоторых основ прокси-сервера Burp Suite, а также нескольких пользовательских и проектных опций.

Прокси

Прокси - это суть Burp Suite. Используя функции прокси-сервера, Burp Suite может видеть весь трафик, проходящий через него. В предыдущих главах мы уже видели, что Burp Suite работает как Человек посередине и помогает нам перехватывать запросы.

Чтобы сделать Burp Suite функциональным, мы должны иметь полную конфигурацию на двух разных концах. Одной из частей является настройка сетевого прокси-сервера в браузере, которую мы видели в предыдущей главе. Другая часть заключается в обеспечении правильной настройки прокси-сервера Burp Suite. По умолчанию, когда мы запускаем Burp Suite, его прокси-сервер прослушивает порт 8080. В этом случае вам больше не нужно выполнять какие-либо настройки. Однако, если порт 8080 уже используется каким-либо другим приложением в вашей системе, может возникнуть конфликт портов, и прокси-служба Burp Suite не запустится. В этом случае вы можете запустить прослушиватель прокси-сервера Burp Suite на любом другом пользовательском порту, который еще не используется. Для этого перейдите на вкладку Прокси, как показано на рис. 3-1.

S. Rahalkar, *A Complete Guide to Burp Suite*,
https://doi.org/10.1007/978-1-4842-6402-7_3

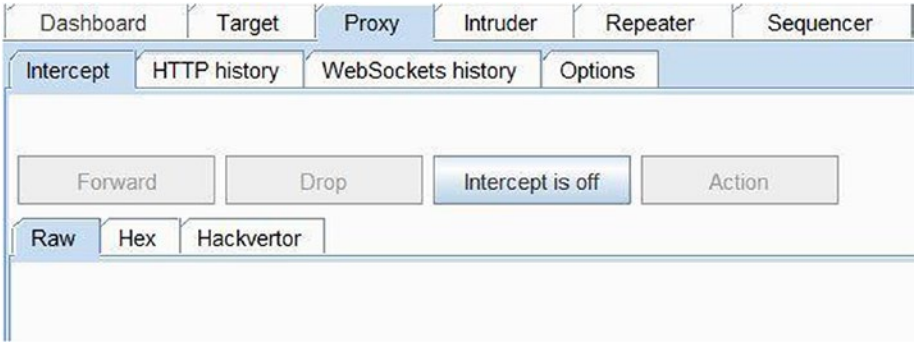

Рис.3-1. *Опция прокси в Burp Suite*

Затем перейдите на вкладку "Options", как показано на рисунке 3-2. Вы заметите, что прокси-сервер Burp Suite по умолчанию работает на localhost и по порту 8080.

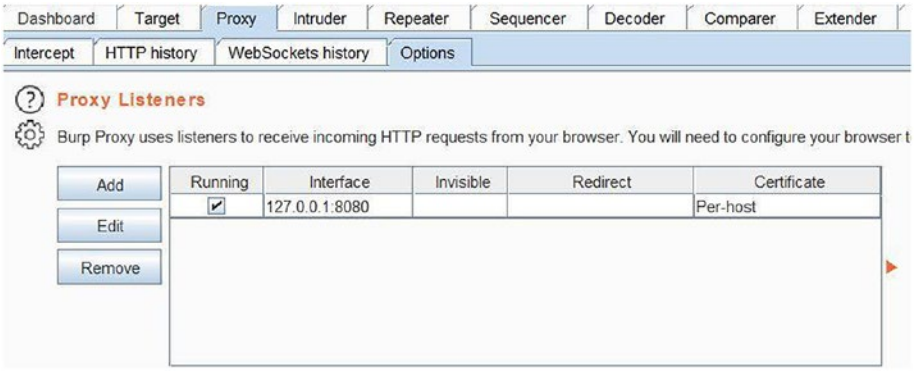

Рис.3-2. *Параметры прокси-сервера*

Чтобы включить прокси-сервер Burp Suite на пользовательском порту, нажмите кнопку "Add", как показано на рисунке 3-3.

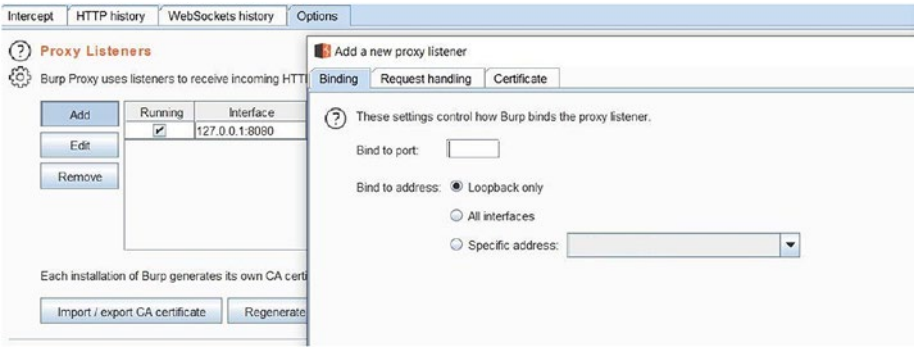

Рис.3-3. *Добавление нового прослушивателя прокси*

Теперь вы можете привязать прокси-сервер Burp Suite к любому пользовательскому порту. После настройки пользовательского порта вы можете оставить адрес привязки по умолчанию как “Loopback only”, или, если в вашей системе имеется несколько сетевых интерфейсов, вы можете выбрать любой из них, используя раскрывающееся меню в опции "Specific address".

Как только прокси-сервер Burp Suite настроен, мы все готовы перехватывать запросы приложений. По умолчанию Burp Suite будет перехватывать только запросы. В любом конкретном сценарии, если вам также требуется перехватывать ответы, это можно сделать с помощью дополнительной настройки. Чтобы включить перехват ответов, перейдите в раздел Proxy > Options и установите флажок “Intercept responses based on the following rules:”, как показано на рисунке 3-4.

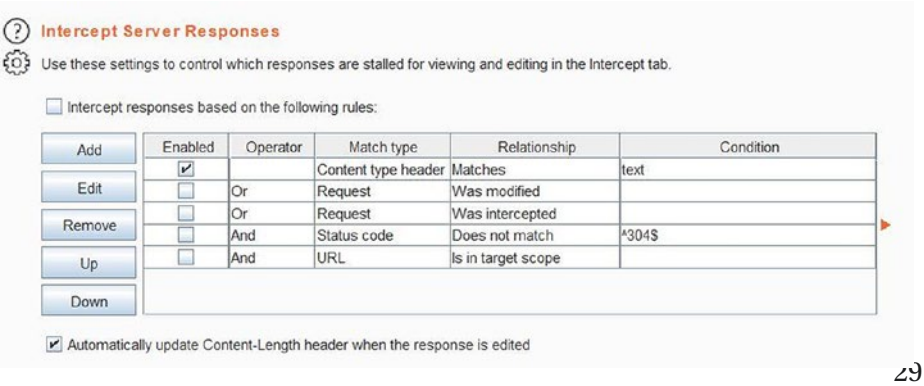

Рис.3-4. *Перехват ответов сервера*

С помощью этой опции вы можете указать различные правила, на основе которых прокси-сервер Burp Suite будет перехватывать ответы.

Теперь, когда у нас есть конфигурация браузера вместе с прокси-сервером Burp Suite, мы можем попытаться перехватить запрос. Перейдите в раздел “Proxy > Intercept” и нажмите “Intercept if off” (чтобы включить перехват). Теперь в браузере введите любой URL-адрес и перейдите на вкладку proxy в Burp Suite, как показано на рис.3-5.

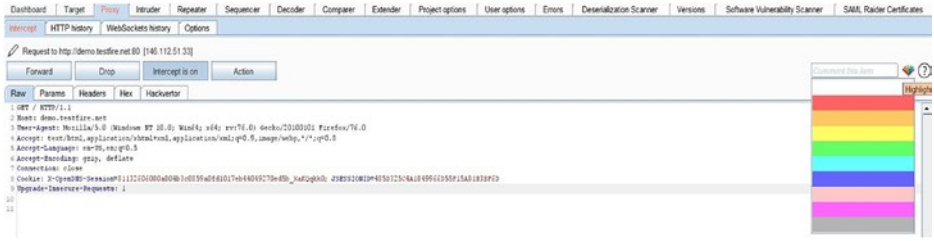

Рис.3-5. *Перехват HTTP-запроса*

Вы заметите, что запрос, который вы сделали из браузера, заблокирован в Burp Suite. Теперь вы можете нажать "Forward", если хотите разрешить отображение URL-адреса в браузере. Вы можете отбросить запрос "Drop", и браузер не загрузит URL-адрес. Если вы отключите опцию перехвата, то все запросы и ответы будут записываться в Burp Suite без какого-либо ручного вмешательства. В правом углу также есть возможность указать цвет запроса, если вы по какой-либо причине хотите выделить его вместе с комментариями.

На вкладке "HTTP History" отображаются все запросы, прошедшие через Burp Suite, как показано на рис.3-6.

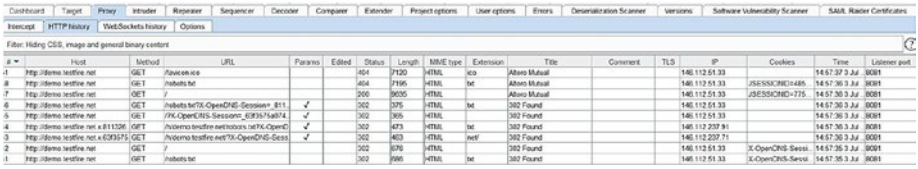

Рис.3-6. *История HTTP-прокси*

История прокси-сервера содержит важную информацию, такую как хост, метод, URL-адрес, параметры (если таковые имеются), статус того, был ли запрос отредактирован/изменен, код состояния HTTP, Длина содержимого, тип MIME, Расширения, Заголовок, IP, файлы cookie и время запроса. Для начала это действительно огромный объем информации. Burp Suite также способен захватывать трафик веб-сокетов по умолчанию, и это можно увидеть на вкладке "Web Sockets history".

Сертификат CA в Burp Suite

Мы уже видели в предыдущей главе, что прокси-сервер Burp Suite работает как "Человек посередине". При доступе к приложению по протоколу HTTPS через Burp Suite прокси-сервер сгенерирует сертификат TLS, подписанный его центром сертификации, и сохранит его в клиентской системе. Чтобы наиболее эффективно использовать Burp Suite в случае HTTPS, рекомендуется загрузить и установить сертификат центра сертификации Burp Suite как доверенный в браузере.

Чтобы импортировать и установить сертификат центра сертификации Burp Suite, сначала убедитесь, что ваш браузер Firefox настроен для работы вместе с прокси-сервером Burp Suite. Затем в адресной строке введите URL “http://burpsuite” как показано на рисунке 3-7.

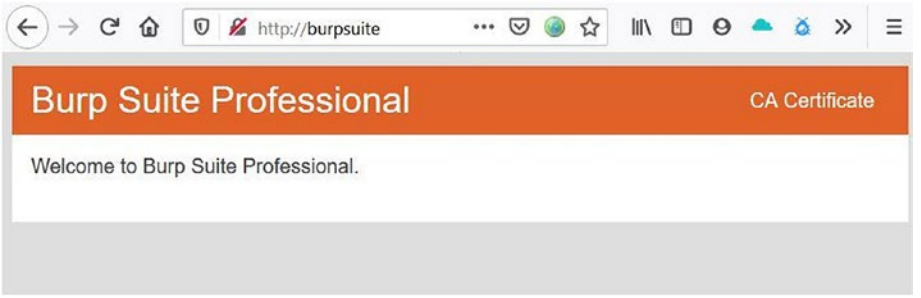

Рис.3-7. *Сертификат CA Burp Suite*

Обратите внимание на правый угол “CA Certificate”. Нажмите на эту опцию, чтобы загрузить файл “cacert.der”, как показано на рисунке 3-8.

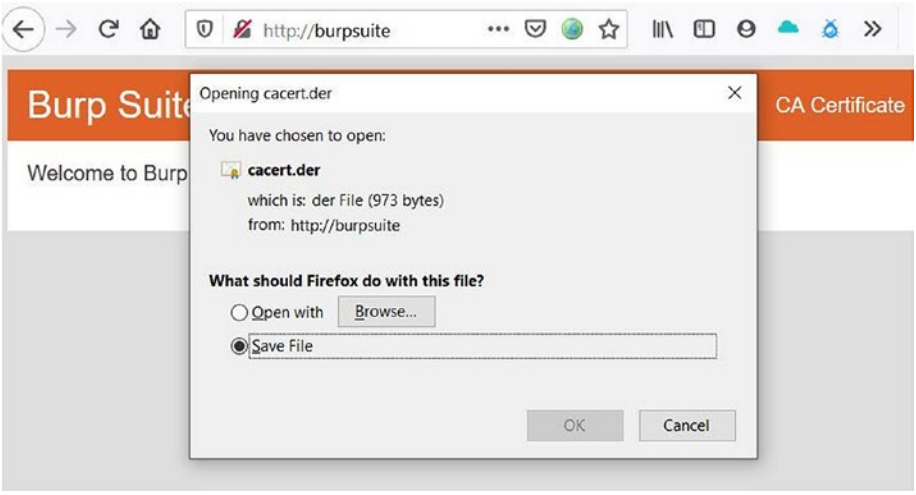

Рис.3-8. *Загрузка сертификата CA Burp Suite*

Затем перейдите в раздел Инструменты Firefox > Параметры и введите "сертификат" в строке поиска, как показано на рис.3-9.

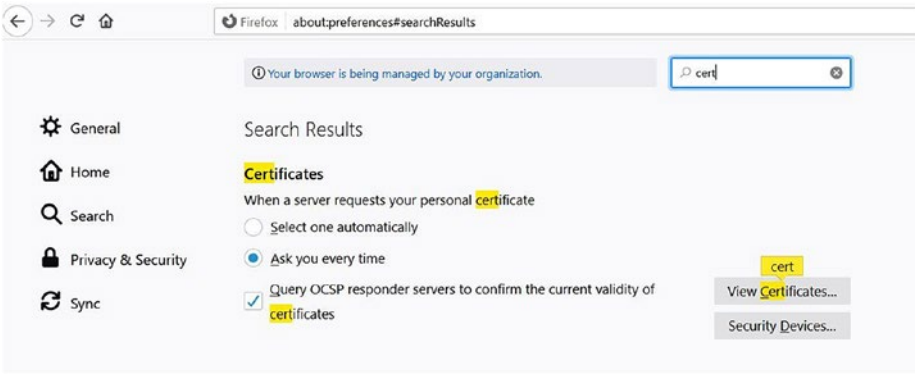

Рис.3-9. *Опция сертификата в Firefox*

Нажмите на опцию "Просмотр сертификатов", а затем используйте кнопку "Импорт", как показано на рис.3-10, чтобы выбрать сертификат Burp Suite, который мы ранее загрузили.

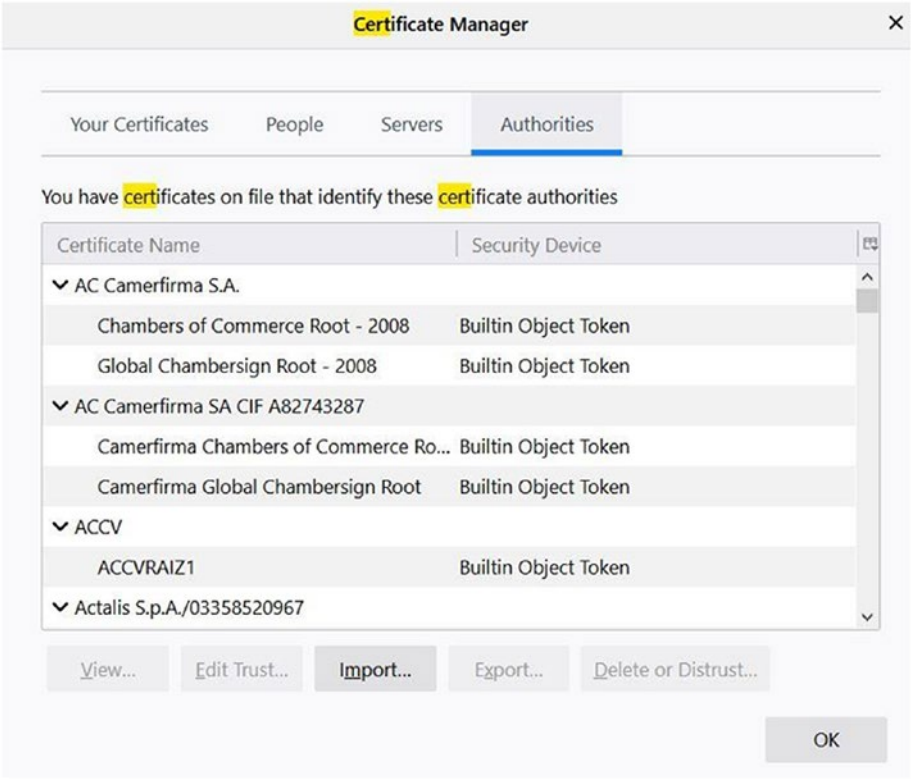

Рис.3-10. *Менеджер сертификатов Firefox*

Аутентификация платформы, вышестоящие прокси-серверы, SOCKS прокси

Аутентификация платформы

Существуют определенные сценарии, в которых приложение, размещенное на целевом веб-сервере, защищено аутентификацией. В таком случае нам необходимо настроить учетные данные в Burp Suite. В отсутствие учетных данных, Burp Suite не сможет получить доступ к защищенной части приложения и пропустить потенциальные проверки. Чтобы настроить аутентификацию платформы, перейдите в раздел “User Options > Connections > Platform Authentication”, как показано на рис. 3-11.

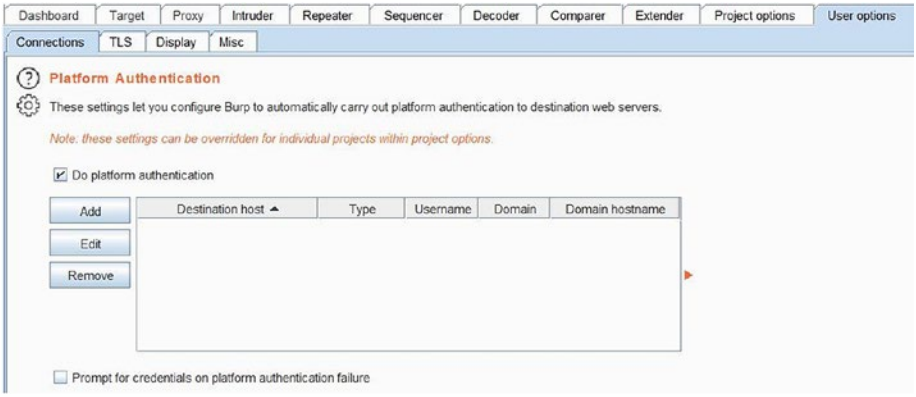

Рис.3-11. *Настройка аутентификации платформы*

Нажмите на кнопку "Add", и появится всплывающее окно, как показано на рис.3-12.

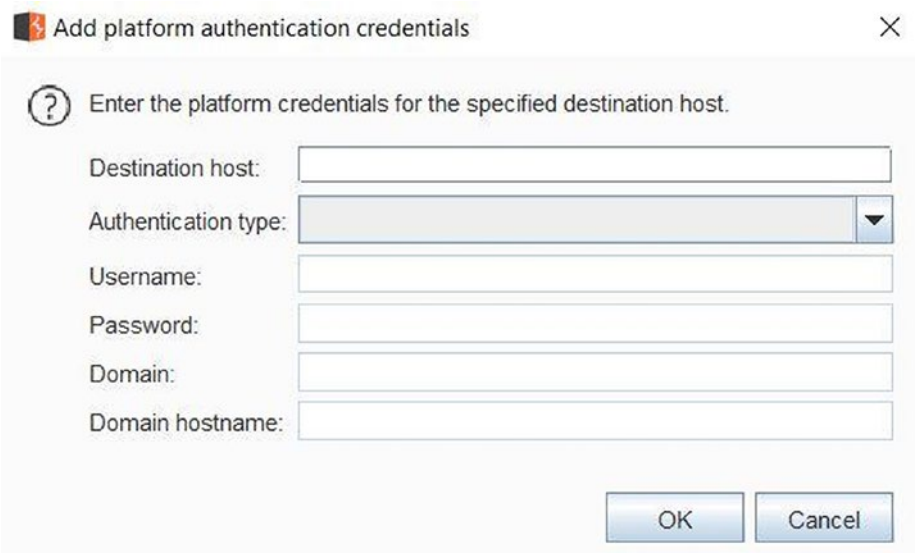

Рис.3-12. *Конфигурация аутентификации платформы*

Нам необходимо настроить конечный хост в виде IP-адреса или имени хоста, типа аутентификации Basic, NTLM V2, NTLM V1, Digest, имени пользователя и пароля, а также Домена, если применимо. Как только эти настройки будут записаны, Burp Suite сможет легко получить доступ к защищенной части приложения с помощью этих учетных данных.

Вышестоящие прокси-серверы

При тестировании приложений в определенных сетевых средах может случиться так, что прямого доступа к целевому приложению не будет. В таком случае нам может потребоваться сначала подключиться к прокси-серверу, а затем подключиться к целевому приложению. Burp Suite позволяет легко настраивать вышестоящие прокси-серверы. Просто перейдите в раздел “User Options > Connections > Upstream Proxy Servers”, как показано на рис.3-13.

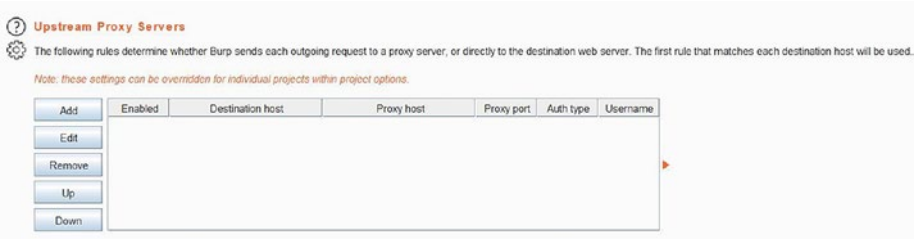

Рис.3-13. *Настройка вышестоящих прокси-серверов*

Нажмите на кнопку "Add", как показано на рис. 3-14, и настройте необходимые параметры прокси-сервера.

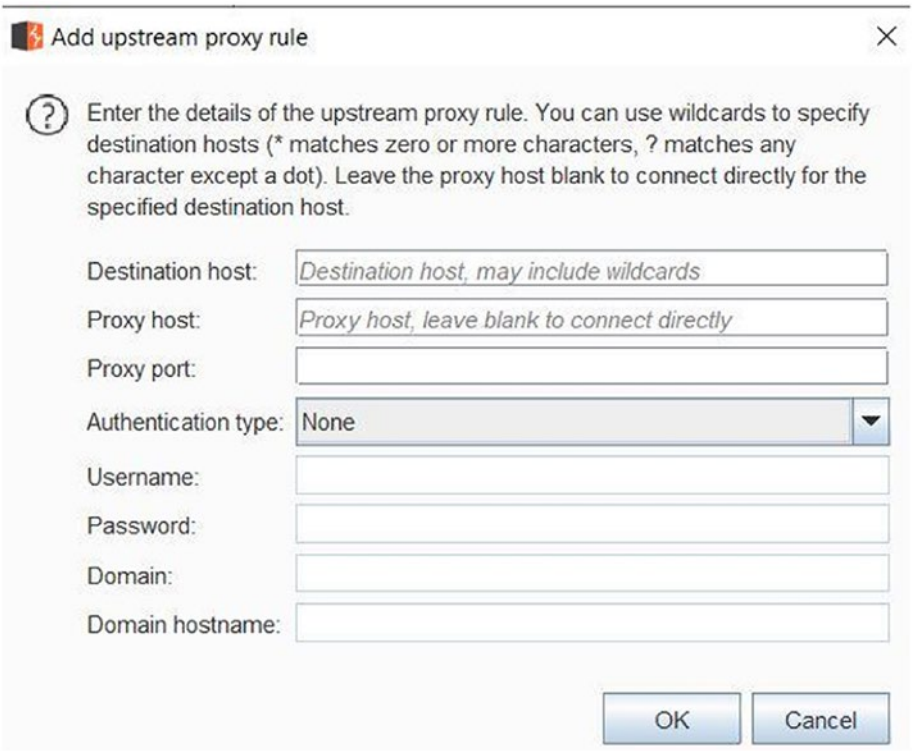

Рис.3-14. *Добавление правила вышестоящего прокси-сервера*

SOCKS Прокси

Burp Suite также позволяет выполнять все запросы на подключение через прокси-сервер SOCKS. Чтобы настроить Burp Suite с прокси-сервером SOCKS, перейдите в раздел "User Options > Connections > SOCKS Proxy", как показано на рис. 3-15, и настройте необходимые параметры прокси-сервера.

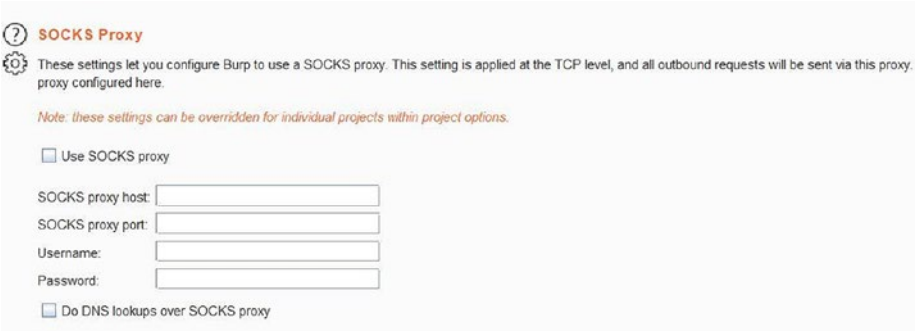

Рис.3-15. *Добавление SOCKS прокси*

Горячие клавиши

Инструмент Burp Suite имеет множество вкладок, инструментов и опций, с которыми мы можем работать. Мы подробно обсудим их в следующих главах. Поначалу инструменты и вкладки Burp Suite могут показаться ошеломляющими. Но все они становятся привычными, когда вы начинаете их использовать. Хотя доступ ко всем этим инструментам и вкладкам можно получить одним нажатием кнопки, но иногда во время работы над проектами гораздо проще использовать сочетания клавиш, чем щелчки мыши.

Инструмент Burp Suite предлагает настройку горячих клавиш, которые являются не чем иным, как сочетаниями клавиш для доступа к определенным инструментам или вкладкам. Чтобы настроить горячие клавиши, перейдите в раздел User Options > Misc > Hotkeys, как показано на рис.3-16.

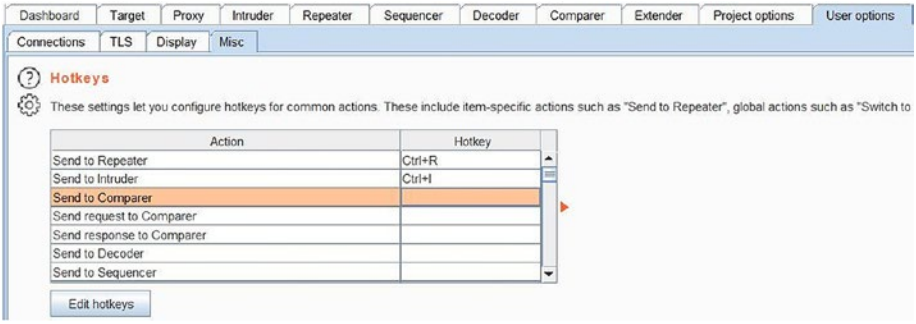

Рис.3-16. *Настройка горячих клавиш в Burp Suite*

По умолчанию горячие клавиши для общих функций в Burp Suite уже настроены. Однако есть опция "Edit hotkeys", чтобы изменить горячие клавиши по умолчанию или настроить горячие клавиши для дополнительных функций.

В таблице 3-1 перечислены некоторые из распространенных горячих клавиш по умолчанию.

Таблица 3-1. Горячие клавиши по умолчанию в Burp Suite

Горячие клавиши	Назначения
Ctrl + R	Отправка в Repeater
Ctrl + I	Отправка в Intruder
Ctrl + F	Переслать перехваченное прокси-сообщение
Ctrl + Shift + T	Переход в Target
Ctrl + Shift + P	Переход в Proxy
Ctrl + Shift + I	Переход в Intruder
Ctrl + Shift + R	Переход в Repeater
Ctrl + Shift + O	Переход в Project Options
Ctrl + Shift + U	URL Декодирование
Ctrl + Shift + B	Base-64 Декодирование
Ctrl + B	Base-64 Кодирование

Помимо горячих клавиш в приведенной выше таблице, все другие стандартные сочетания клавиш для выбора всего текста, вырезания, копирования и вставки работают стандартным способом.

Резервное копирование проекта

При работе над проектами с использованием Burp Suite генерируются большие объемы данных в виде запросов и ответов. Возникает необходимость сохранять копию этих данных через регулярные промежутки времени. Вместо того, чтобы выполнять эту задачу вручную, Burp Suite предлагает функцию автоматического создания резервной копии данных через определенные промежутки времени.

Чтобы включить автоматическое резервное копирование, перейдите в раздел User Options Misc Automatic Project Backup, как показано на рис.3-17.

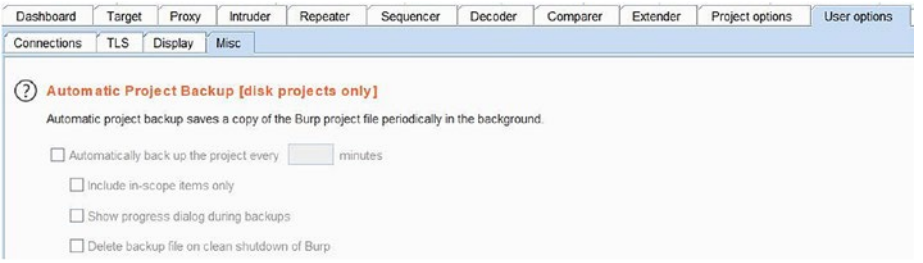

Рис.3-17. *Настройка автоматического резервного копирования проектов*

Используя эту функцию, мы можем указать продолжительность в минутах, по истечении которой Burp Suite автоматически запустит резервное копирование.

Rest API

Хотя мы используем Burp Suite в основном для ручного тестирования безопасности приложений, существует множество других инструментов и вариантов использования, которые необходимо использовать вместе с Burp Suite. Могут существовать другие средства безопасности, которые необходимо интегрировать с Burp Suite, или определенные пользовательские сценарии автоматизации, которые также должны автоматически запускать действия в Burp Suite.

Для всех таких целей Burp Suite предоставляет пользователям интерфейс REST API. REST означает Передачу репрезентативного состояния, а API - интерфейс прикладного программирования. REST API - это самый популярный способ подключения различных приложений.

Чтобы включить REST API в Burp Suite, перейдите в раздел “User Options > Misc”, как показано на рис. 3-18.

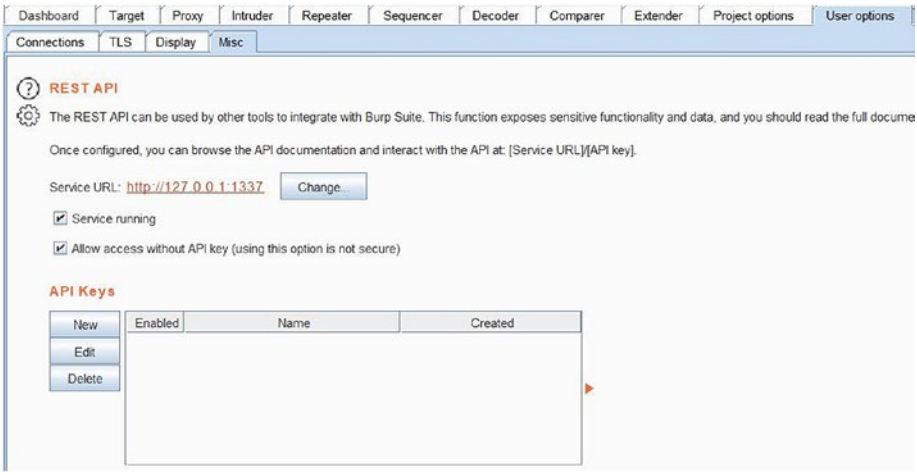

Рис.3-18. *Настройка интерфейса REST API в Burp Suite*

Чтобы включить REST API в Burp Suite, просто установите флажок “Service running”. REST API будет доступен по умолчанию по http://127.0.0.1:1337 как показано на рисунке 3-19.

Рис.3-19. *Интерфейс REST API в Burp Suite*

Интерфейс REST API содержит список всех поддерживаемых глаголов или методов, вызываемых конечных точек, передаваемых параметров и ожидаемых ответов. Теперь это полностью зависит от конкретного сценария или варианта использования, от того, как можно использовать этот интерфейс REST API.

Обратная связь по производительности

Как и в случае с любым другим инструментом, Burp Suite имеет возможность собирать и отправлять определенные диагностические данные, которые могут быть полезны для повышения производительности Burp Suite. Это совершенно необязательная функция, и если она включена, она собирает данные только о внутреннем функционировании, а не о конкретных пользователях или данных проекта. Burp Suite также предоставляет функции для регистрации всех исключений или для сообщения о конкретной ошибке команде Burp Suite.

Чтобы использовать параметры обратной связи по производительности, перейдите в раздел User Options > Misc > Performance Feedback, как показано на рис.3-20.

Рис.3-20. *Настройка обратной связи по производительности*

Параметры проекта

Эти параметры включают разрешение имен хостов, Запросы вне области действия, Перенаправления, Конфигурация TLS, Правила обработки сеансов, Файлы cookie и Макросы.

Таймауты

Обработка запросов и ответов - это основная функциональность Burp Suite. Возможны такие сценарии, как сбой целевого приложения или проблемы с подключением, при которых Burp Suite необходимо решить, как долго он должен ждать ответа на запрос, прежде чем отправлять его. Эти настройки определяются значениями времени ожидания. Они настроены по умолчанию и могут быть оставлены нетронутыми, если нет явной необходимости это изменить. Для изменения значений тайм-аута по умолчанию перейдите в раздел “Project Options > Connections > Time Outs”, как показано на рис. 3-21.

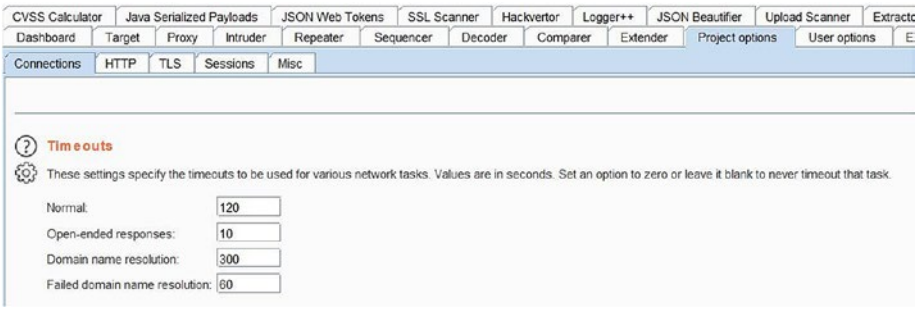

Рис.3-21. *Настройка тайм-аута запроса*

Разрешения имен хостов

Разрешение имени хоста обычно происходит с помощью либо локального файла хоста, либо сетевого DNS. Тем не менее, Burp Suite допускает пользовательские разрешения имени хостов. Это может быть полезно в конкретных сценариях, когда к приложению, размещенному во внутренней сети, необходимо получить доступ с использованием определенного имени хоста или URL-адреса.

Чтобы указать пользовательские правила разрешения имен хостов, перейдите в раздел “Project Options > Connections > Hostname Resolutions”, как показано на рис.3-22.

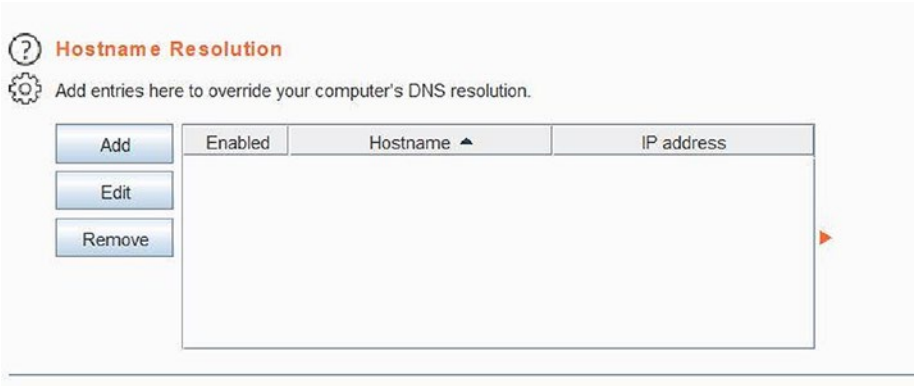

Рис.3-22. *Настройка разрешения имени хоста*

Нажмите на кнопку "Add', а затем вы получите всплывающее окно для ввода пользовательского имени хоста и связанного с ним IP-адреса, как показано на рис.3-23.

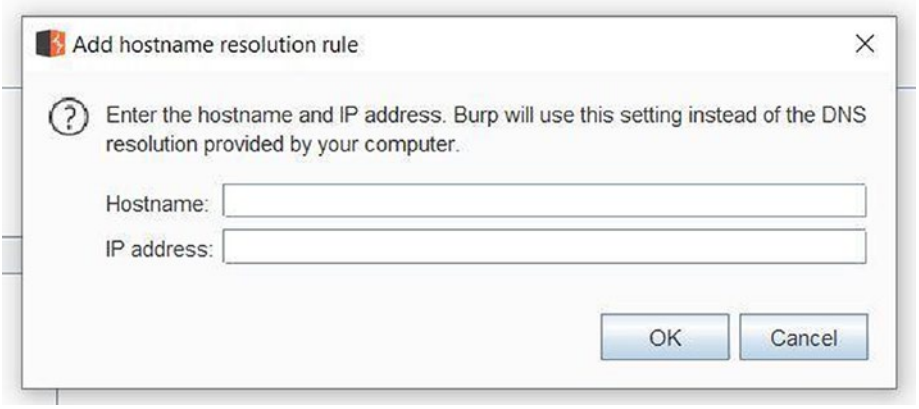

Рис.3-23. *Добавление правила разрешения имен хостов*

Запросы выходящие за рамки

Как только браузер настроен для работы вместе с Burp Suite, Burp Suite по умолчанию будет захватывать весь HTTP-трафик на всех вкладках. Этот трафик может быть подавляющим и отвлекающим. Из всего захваченного трафика нам нужно сосредоточиться только на требуемой цели, которую мы тестируем. Этого можно достичь, используя раздел, который мы рассмотрим в следующей главе. Burp Suite предоставляет функцию, позволяющую просто отбрасывать все запросы, которые выходят за рамки. Это можно сделать, перейдя в раздел “Project Options > Connections > Out-of-scope Requests”, как показано на рис.3-24.

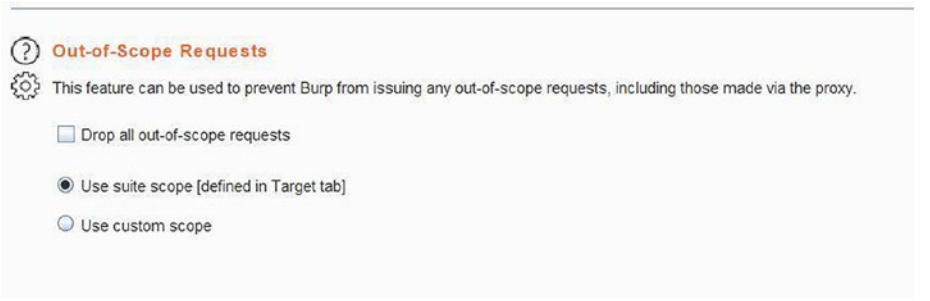

Рис.3-24. *Настройка правил для запросов, выходящих за рамки области действия*

Выбрав опцию "Use custom scope", мы можем добавить URL, который мы хотим исключить из области действия и отбросить прокси, как показано на рис.3-25.

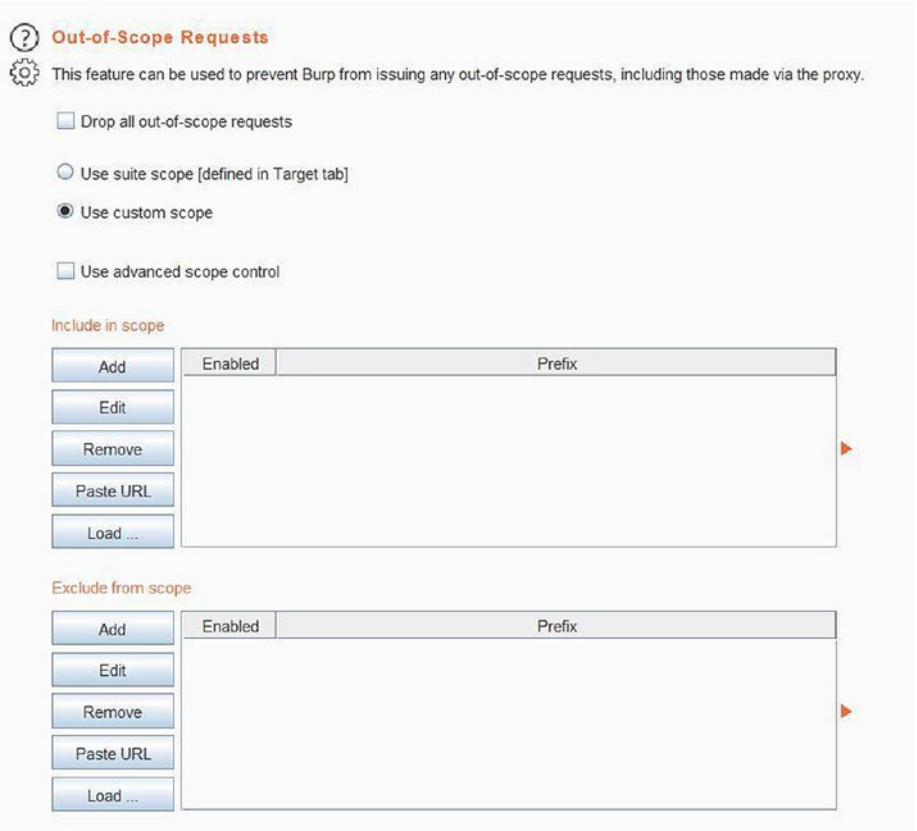

Рис.3-25. *Определение правил для запросов, выходящих за рамки*

Перенаправление

Автоматическое сканирование приложений требует обработки HTTP-перенаправлений. Механизм сканирования должен принять меры, как только он обнаружит перенаправление любой страницы. В Burp Suite по умолчанию настроены правила перенаправления, и их можно оставить нетронутыми, если только нет явной необходимости их изменять или нет необходимости в дополнительной настройке перенаправлений, управляемых JavaScript. Для настройки правил перенаправления перейдите в раздел “Project Options > HTTP > Redirections”, как показано на рис. 3-26.

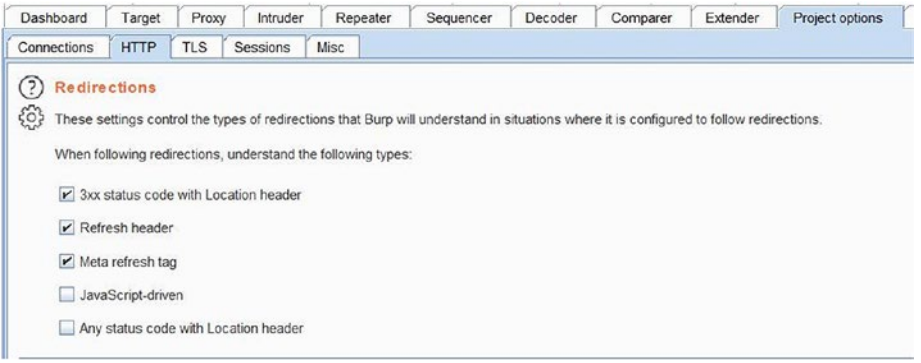

Рис.3-26. *Конфигурация перенаправлений*

Cookie Jar

Любой инструмент сканирования / тестирования приложений должен поддерживать хранилище файлов cookie, которые он будет использовать для управления текущими сеансами приложений. Возможность обнаружения во время сеанса особенно необходима при выполнении автоматического сканирования. Burp Suite хранит файлы cookie приложения в контейнере под названием “Cookie Jar”. По умолчанию Cookie Jar отслеживает прокси-трафик для извлечения и хранения любых файлов cookie; однако мы можем указать Burp Suite отслеживать и захватывать файлы cookie с помощью других инструментов, таких как Scanner, Repeater, Intruder, Sequencer, и Extender. Это можно сделать, перейдя в раздел “Project Options > Sessions > Cookie Jar”, как показано на рис.3-27.

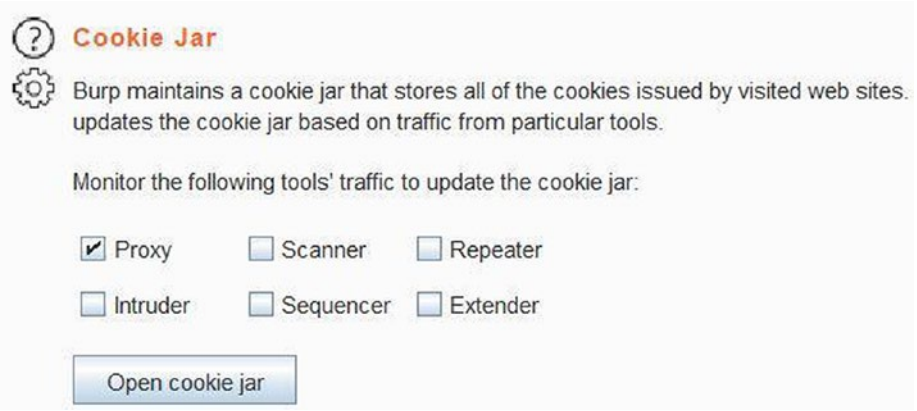

Рис.3-27. *Настройка параметров Cookie Jar*

Макросы

В процессе тестирования безопасности приложения может потребоваться многократное выполнение определенной последовательности действий. Burp Suite предоставляет отличную функциональность макросов для достижения этой цели. Редактор макросов доступен в разделе “Project Options > Sessions > Macros”, как показано на рис.3-28. Вы можете просто нажать на кнопку "Add" и следовать инструкциям мастера для записи действий.

Рис.3-28. *Конфигурация макросов*

Резюме

В этой главе мы узнали о настройке прокси-сервера Burp Suite вместе с сертификатом CA. Затем мы рассмотрели несколько вариантов, таких как аутентификация платформы, вышестоящий прокси-сервер, socks прокси и т.д. Затем мы рассмотрели несколько других полезных конфигураций, включая горячие клавиши, резервное копирование проектов, использование API в Burp Suite, параметры проекта, разрешения имен хостов, область действий и перенаправления.

В следующей главе мы рассмотрим дашборд в Burp Suite, вкладку "Цель" и "инструменты взаимодействия".

Упражнения

- Настройте свой любимый браузер и Burp Suite для работы с пользовательским прокси-портом.
- Попробуйте установить сертификат CA Burp Suite для Chrome, Edge и Opera.
- Просмотрите список горячих клавиш по умолчанию и попробуйте настроить дополнительные сочетания клавиш по вашему выбору.

Глава 4

Дашборд, цель и инструменты взаимодействия

В предыдущей главе мы познакомились с некоторыми основами настройки прокси-сервера, параметрами пользователя и параметрами проекта. В этой главе мы начнем с ознакомления дашборда Burp Suite, инструментами цели и взаимодействия.

Дашбоард

Дашборд, как следует из названия, является той важной частью Burp Suite, которая, по сути, обобщает различные действия и задачи, выполняемые в разных компонентах. На рис. 4-1 показан типичный вид Дашборд в Burp Suite.

S. Rahalkar, *A Complete Guide to Burp Suite*,
https://doi.org/10.1007/978-1-4842-6402-7_4

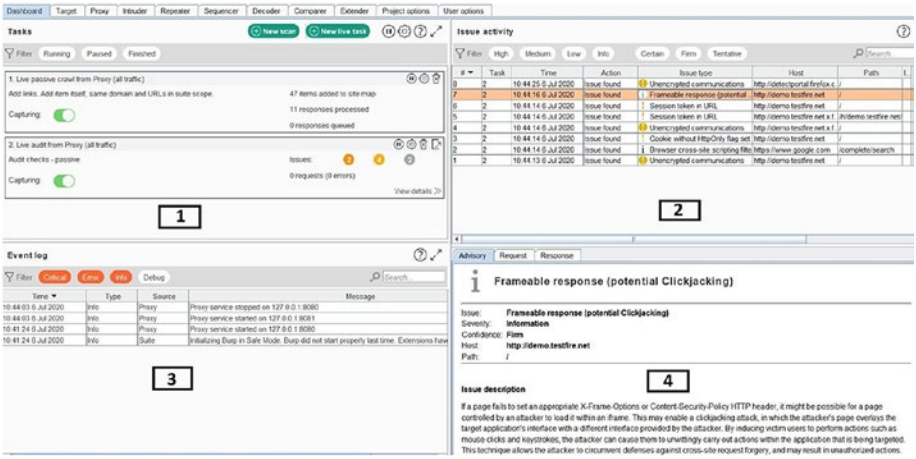

Рис.4-1. *Дашборд в Burp Suite*

Для лучшего понимания мы разделим Дашборд на четыре части и попытаемся подробно объяснить каждую из них. В учебных целях ознакомьтесь с цифрами от 1 до 4 на рис. 4-1.

В первой части дашборда отображаются данные о текущих сканированиях, как показано на рисунке 4-2. Сканирование может быть как пассивным, так и активным. Если выполняется несколько сканирований, мы можем отфильтровать их в зависимости от их состояния: запущено, приостановлено или завершено. Мы также видим сводку высокого уровня о проблемах, обнаруженных при сканировании. Также есть возможность создать новую задачу сканирования, которую мы рассмотрим отдельно в следующей главе.

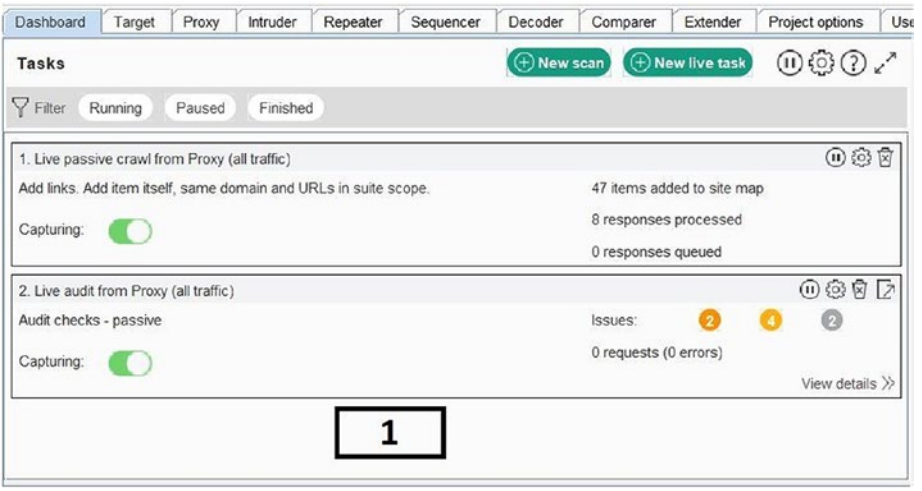

Рис.4-2. *Задачи в дашборте Burp Suite*

Вторая часть дашборда выглядит так, как показано на рисунке 4-3. Может случиться так, что выполняется либо пассивное сканирование, либо активное сканирование, и в любом случае в этом разделе выделен список обнаруженных уязвимостей. В этом разделе также освещается такая информация, как время обнаружения проблемы, тип проблемы, хост или цель, на которой была обнаружена проблема, уязвимый URL-адрес, серьезность проблемы и уровень достоверности. Burp Suite может отмечать разные уровни достоверности для разных проблем на основе ответов; однако проблемы требуют ручной проверки, чтобы убедиться в их достоверности.

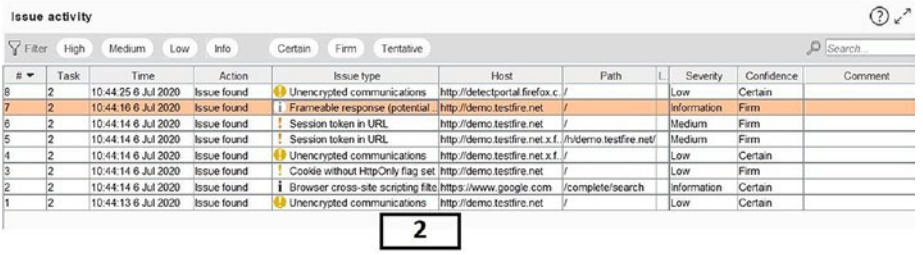

Рис.4-3. *Активность проблем в дашборде Burp Suite*

Третья часть дашборда содержит краткое описание функциональных событий Burp Suite, как показано на рисунке 4-4. В основном это состояние прокси-службы, сбои подключения TLS (если таковые имеются), сбои аутентификации, тайм-ауты и т.д. Например, если в вашей системе уже настроен порт 8080 с какой-либо другой службой, в этой части дашборда будет выделено, что прокси-служба не может быть запущена по порту 8080. Или если по какой-либо причине прокси-служба остановится, то она также будет выделена здесь. В целом, это дает представление о том, правильно ли работает прокси-сервер Burp Suite и связанные с ним службы или нет.

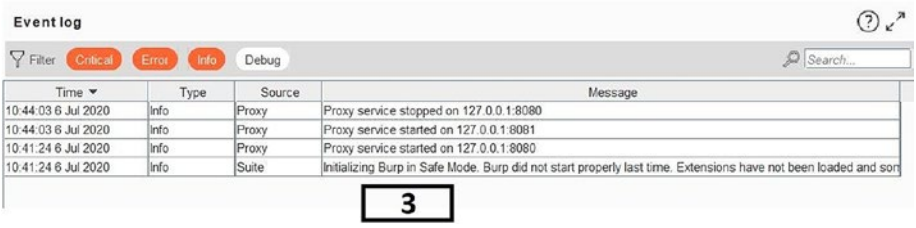

Рис.4-4. *Журнал событий на дашбоард в Burp Suite*

Последняя часть дашборда, как показано на рис. 4-5, выделяет детали проблемы. Если вы хотите просмотреть подробную информацию о любой из проблем, выделенных во время пассивного или активного сканирования, просто нажмите на эту проблему, как показано на рис.4-3, и информация будет доступна соответствующим образом. Сведения о проблеме включают полное описание проблемы и рекомендации по исправлению, а также фактический запрос и ответ.

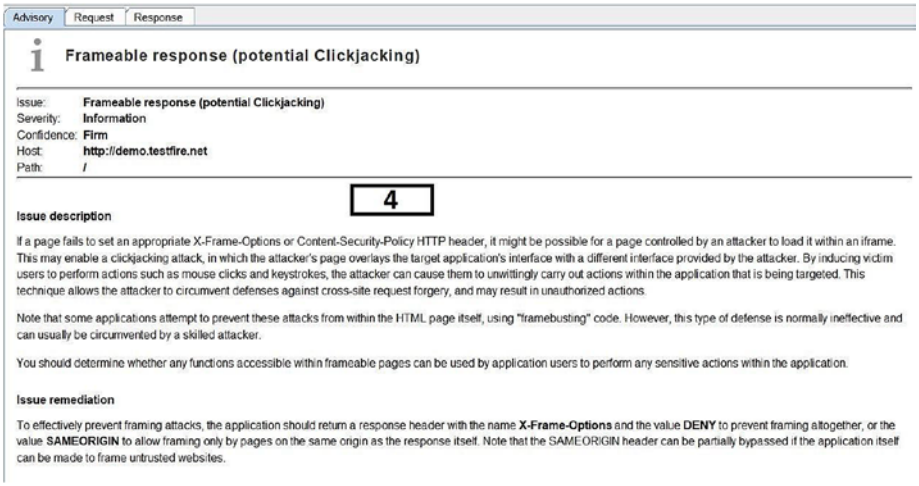

Рис.4-5. *Подробная информация о проблеме на дашборте в Burp Suite*

Целевая вкладка

Как и дашборд Burp Suite, которую мы видели в предыдущем разделе, вкладка "Target" является не менее важной рабочей областью. Вкладка "Target" также содержит несколько панелей, как показано на рис.4-6.

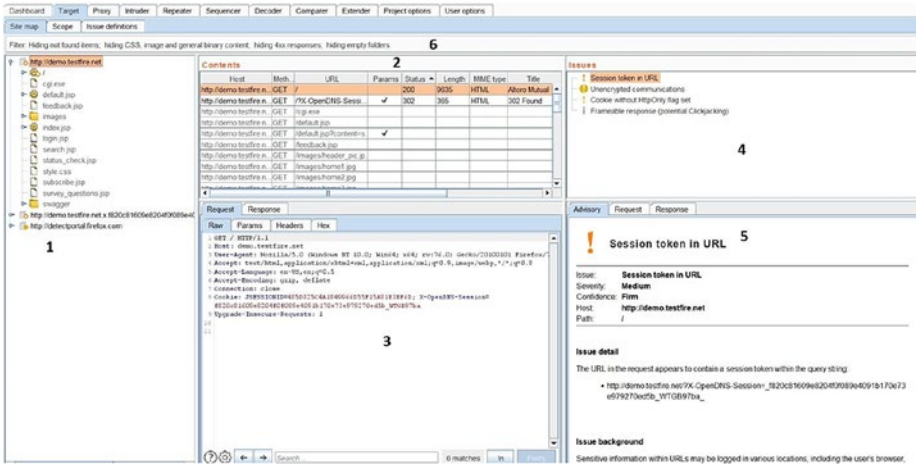

Рис.4-6. *Вкладка "Target" в Burp Suite*

Чтобы понять различные части целевой вкладки, давайте рассмотрим по одной части за раз, ссылаясь на цифры от 1 до 6, как показано на рис.4-6.

Крайний левый раздел на целевой вкладке с номером "1" на рисунке 4-6 показан на рисунке 4-7. В этом разделе создается иерархия сайта, который мы просматриваем. В нем перечислены все конечные узлы, папки и т.д. В виде четко определенной древовидной структуры. Это помогает получить представление о том, насколько большим может быть сайт / приложение или что общего у его содержимого. Это очень похоже на карту сайта.

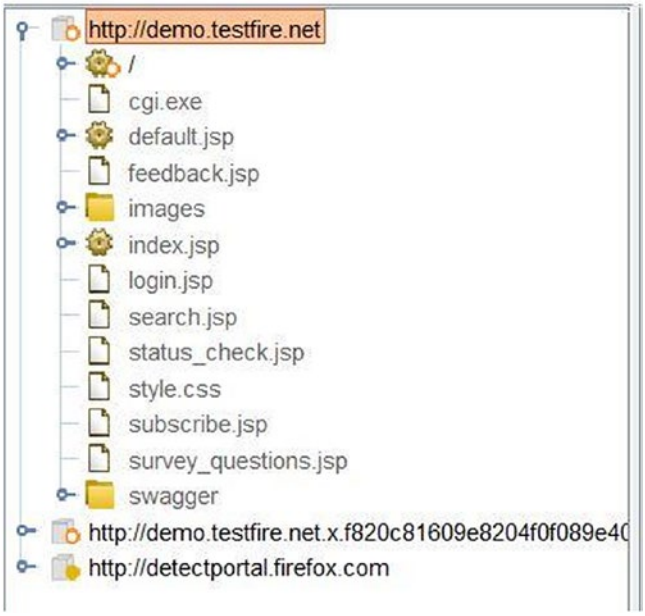

Рис.4-7. *Карта приложений / иерархия*

Следующий раздел, пронумерованный как "2" в соответствии с рисунком 4-6, показан на рисунке 4-8. В этом разделе перечислены все HTTP-запросы, которые были сделаны, а также другие сведения, такие как точный хост, используемый метод HTTP, целевой URL-адрес, были ли у URL какие-либо параметры, код ответа статуса HTTP, длина содержимого, тип MIME и заголовок страницы, если таковые имеются. Простой просмотр этого раздела может помочь вам найти интересные URL-адреса, особенно те, у которых есть параметры для ввода.

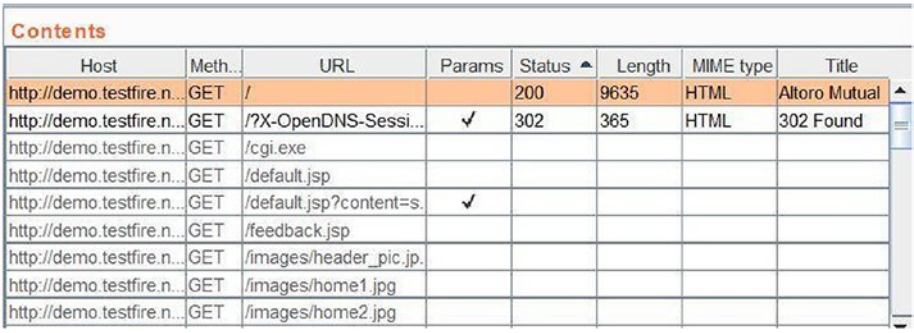

Рис.4-8. *Список запросов*

Следующий раздел, пронумерованный как "3" в соответствии с рисунком 4-6, показан на рисунке 4-9. В этом разделе показаны актуальные HTTP-запросы и ответы. Вы можете просмотреть необработанный запрос по умолчанию, но также просмотреть запрос в форме параметров. Вы также можете увидеть все заголовки, используемые в ответе.

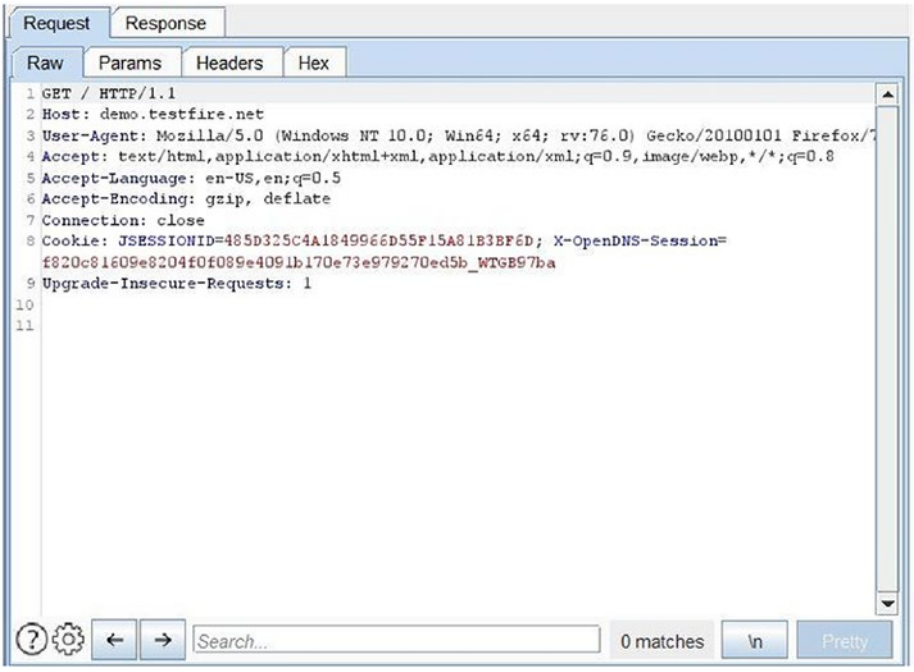

Рис.4-9. *Окно просмотра запросов и ответов*

Следующий раздел, пронумерованный как "4" в соответствии с рисунком 4-6, показан на рис.4-10. В этом разделе показан список проблем, обнаруженных во время пассивного или активного сканирования. Проблемы классифицируются в соответствии с серьезностью, причем проблемы с наибольшей серьезностью отображаются вверху.

Рис.4-10. *Проблемы, обнаруженные в целевом приложении*

Следующий раздел, пронумерованный как " 5 " в соответствии с рисунком 4-6, показан на рис. 4-11. В этом разделе показаны сведения о проблеме для любой из выбранных проблем. Он содержит описание проблемы вместе с рекомендациями по исправлению. В этом разделе также можно просмотреть фактический запрос и ответ, на основе которых была выделена проблема.

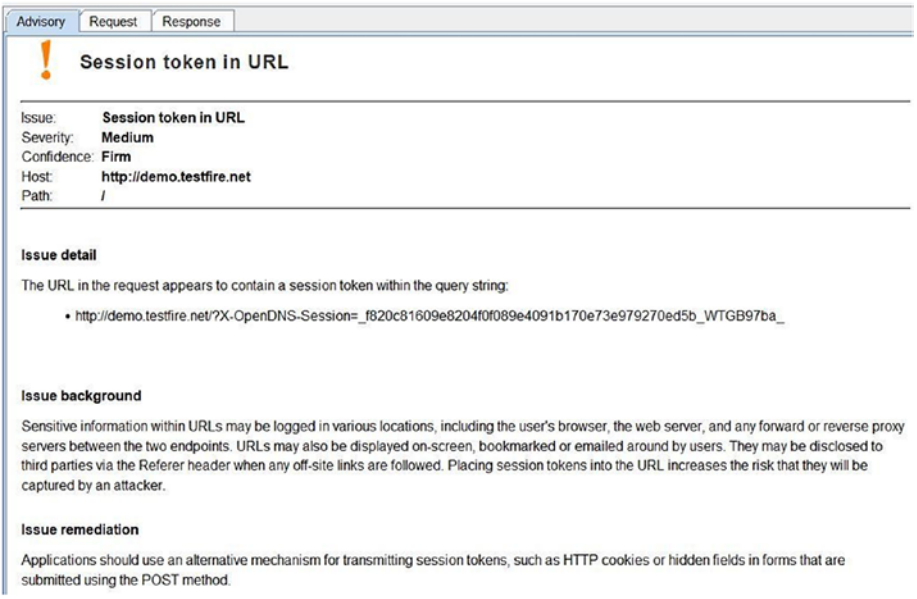

Рис.4-11. *Детали проблемы*

Следующий раздел, пронумерованный как " 6 " в соответствии с рисунком 4-6, показан на рис. 4-12. Как только мы настроим наш браузер для работы вместе с Burp Suite, может быть захвачено много трафика. Следовательно, для фильтрации только необходимых данных можно использовать несколько фильтров. Некоторые распространенные и полезные фильтры включают фильтрацию по типу запроса, типу MIME, коду состояния, расширению и т.д. Эта функция также позволяет выполнять поиск определенного элемента в данных, собранных через прокси.

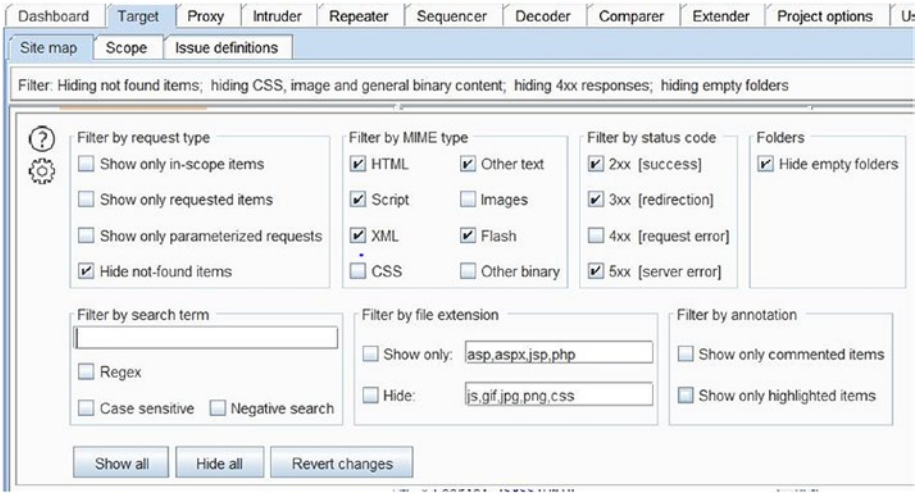

Рис.4-12. *Фильтры карты сайта*

Инструменты взаимодействия

Инструменты взаимодействия - это не что иное, как небольшие утилиты, которые помогают выполнять некоторые дополнительные задачи в Burp Suite. В этом разделе мы рассмотрим несколько таких инструментов взаимодействия, служащих различным целям. Чтобы получить доступ к инструментам взаимодействия, просто щелкните правой кнопкой мыши по целевому URL-адресу, по которому вы хотите запустить инструменты взаимодействия, как показано на рис.4-13.

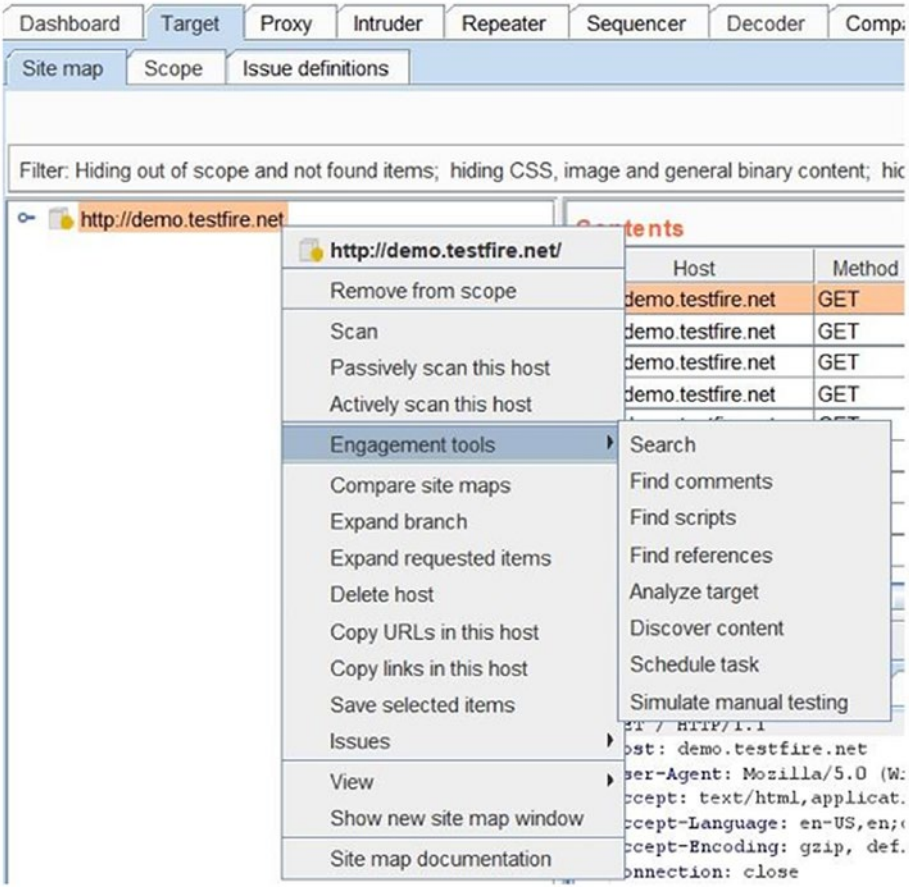

Рис.4-13. *Инструменты взаимодействия в Burp Suite*

Первый инструмент взаимодействия - это простой поиск (Search), как показано на рис. 4-14. Это позволяет пользователям искать любое ключевое слово в запросах или ответах от выбранной цели.

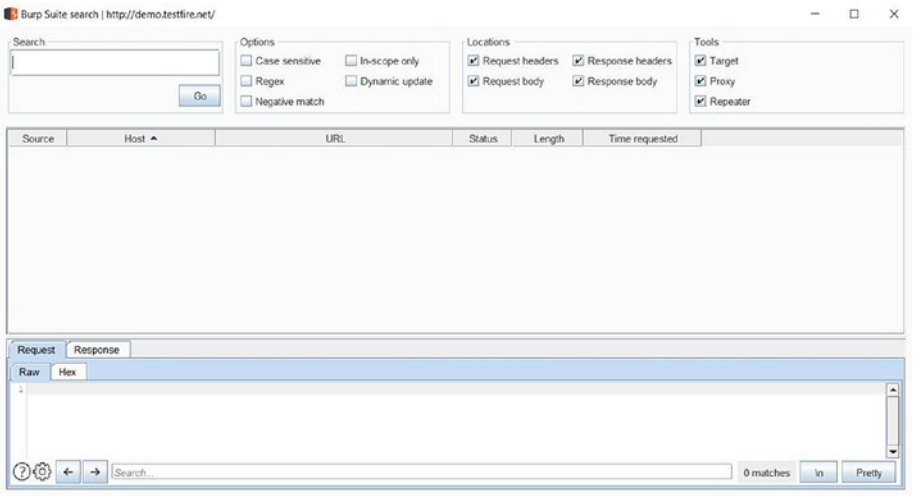

Рис.4-14. *Инструмент поиска*

Следующим инструментом взаимодействия является "Поиск комментариев" (Find comments), как показано на рис.4-15. Это просто просматривает собранные данные и находит любые комментарии к коду. Стоит просмотреть эти комментарии, так как всегда есть возможность найти что-то чувствительное в комментариях.

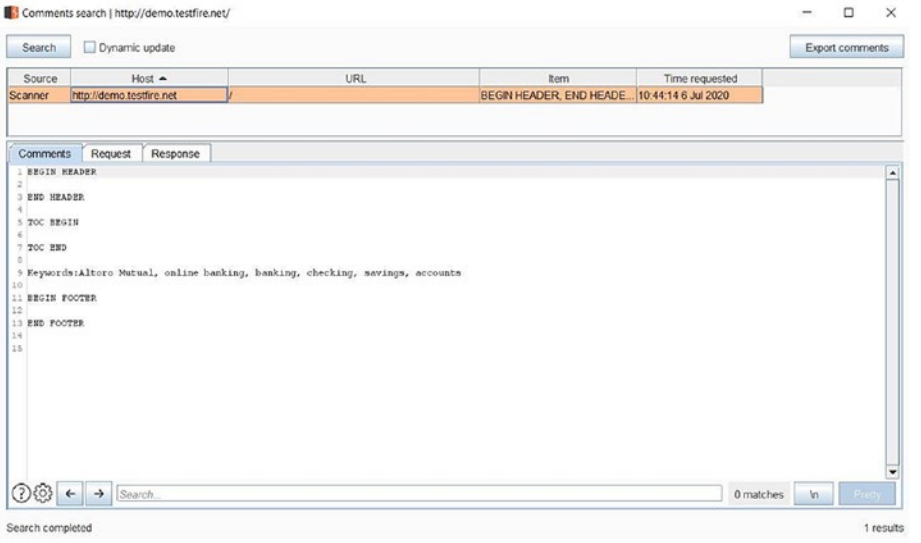

Рис.4-15. *Инструмент поиска комментариев*

Следующим инструментом взаимодействия является "Поиск сценариев" (Find scripts), как показано на рис.4-16. Этот инструмент просто выполняет поиск всех сценариев в пределах выбранной цели.

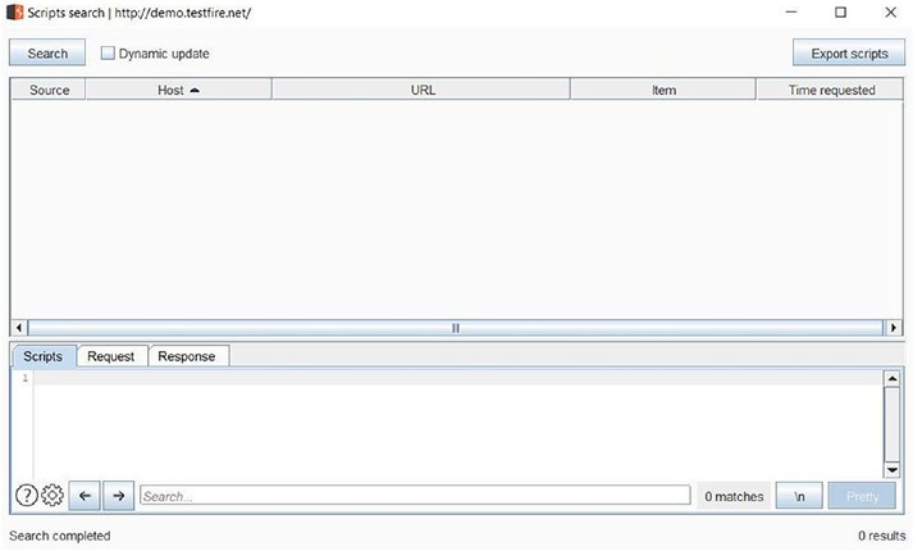

Рис.4-16. *Инструмент поиска сценариев*

Следующим инструментом взаимодействия является "Поиск ссылок" (Find references), как показано на рис. 4-17. Этот инструмент перечисляет все компоненты Burp Suite, в которых он может найти любую ссылку на выбранную цель. Например, URL-адрес "demo.testfire.net" появилась в сканере, а также в качестве цели в Burp Suite, как показано на рис.4-17.

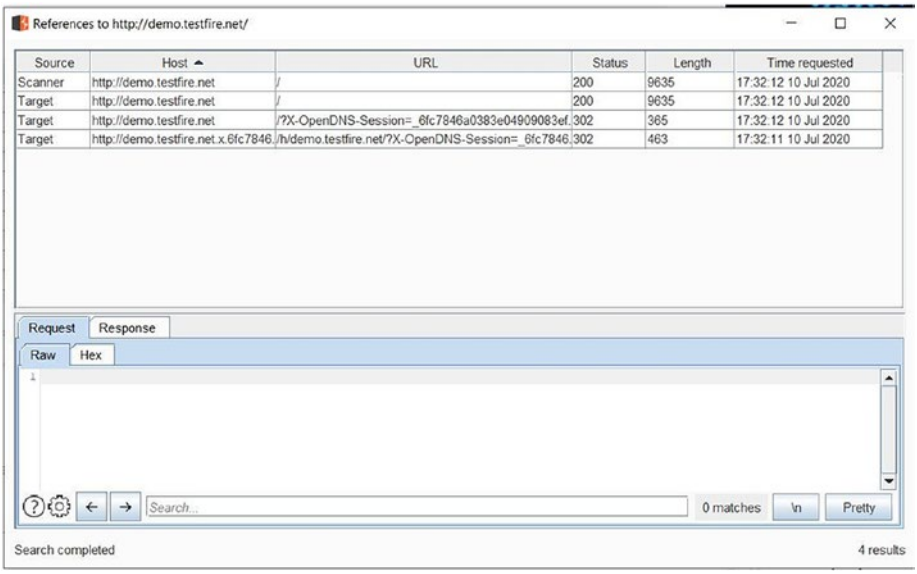

Рис.4-17. *Инструмент поиска ссылок*

Следующим инструментом взаимодействия является "Анализатор целей" (Target Analyzer), как показано на рис. 4-18. Эта утилита предоставляет статистику в основном по размеру приложения с точки зрения количества динамических URL-адресов, количества статических URL-адресов, количества параметров и т.д. Эти статистические данные помогают тестировщику получить оценку усилий, которые потребуются для тестирования приложения.

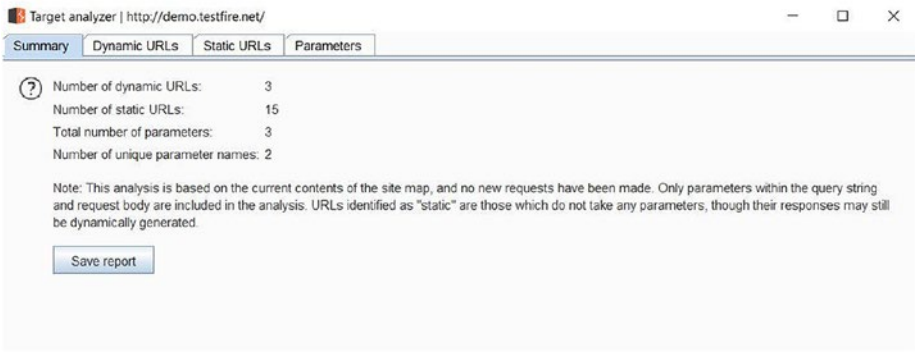

Рис.4-18. *Целевой анализатор*

Следующим инструментом взаимодействия является "Обнаружение контента" (Content Discovery), как показано на рис.4-19. Этот инструмент помогает определить правила спайдеринга или обхода. Например, это помогает определить правила спайдеринга при обнаружении новой цели, например, какая длина и ширина ее должна быть спайдеризована или следует также использовать только файлы или каталоги. Это создает хорошо структурированную карту сайта. Это необязательно, и его можно оставить по умолчанию.

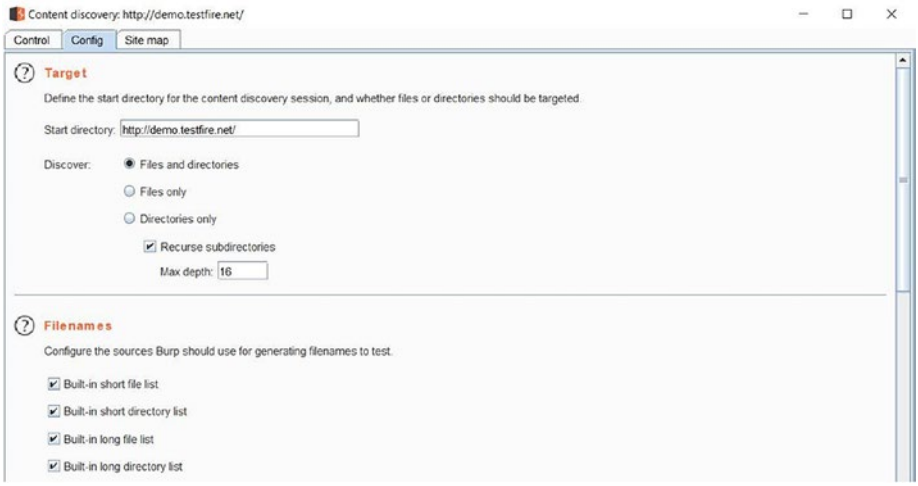

Рис.4-19. *Инструмент обнаружения контента*

Следующим инструментом взаимодействия является "Задание по расписанию" (Schedule Task), как показано на рис. 4-20. Это простая утилита для планирования любых задач Burp Suite. Используя это, вы можете либо приостановить, либо возобновить выполнение какой-либо задачи.

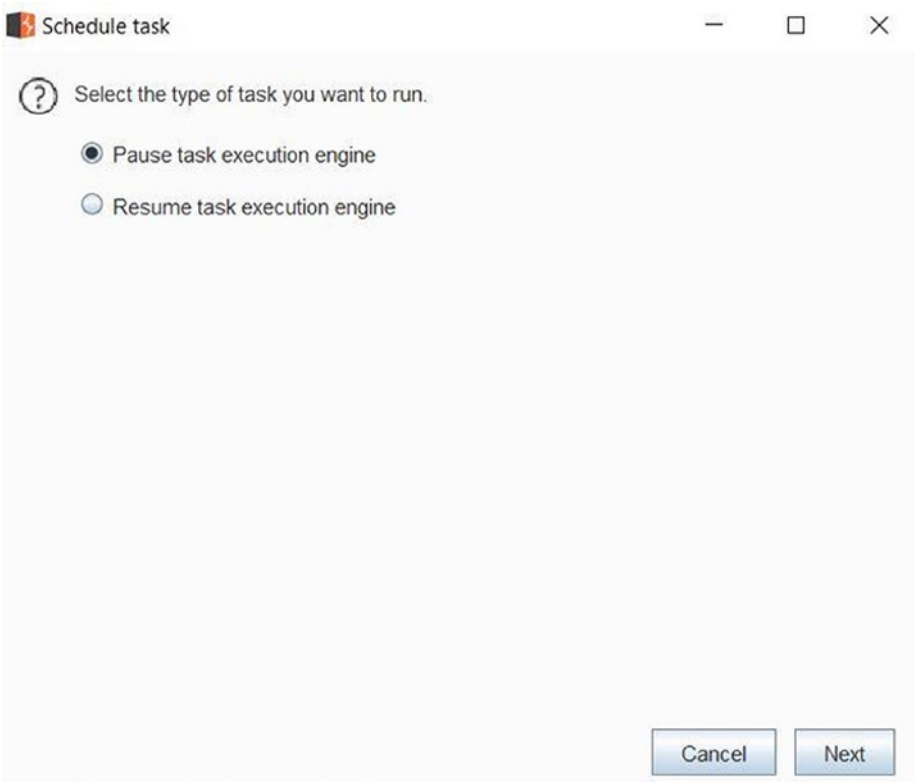

Рис.4-20. *Планировщик задач в Burp Suite*

Следующим инструментом взаимодействия является "Симулятор ручного тестирования" (Manual Testing Simulator), как показано на рис.4-21. Этот инструмент будет отправлять HTTP-запросы цели через случайные промежутки времени и может помочь поддерживать сеанс. Это может быть полезно в сценарии, когда тестировщик находится в перерыве, и существует вероятность того, что время сеанса приложения истечет из-за бездействия.

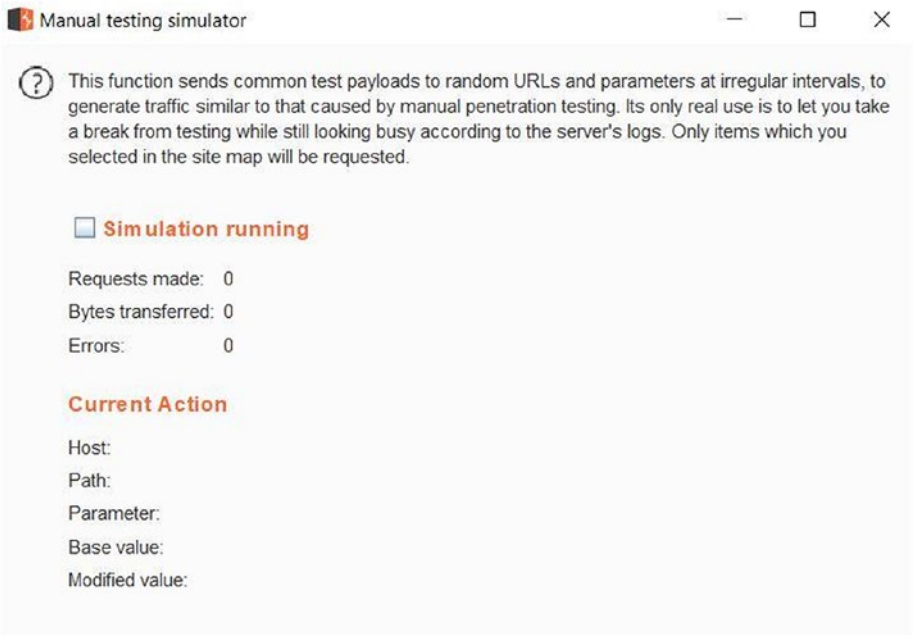

Рис.4-21. *Симулятор ручного тестирования*

Резюме

В этой главе мы изучили основы дашборда Burp Suite, вкладки "Цель" (Target), а также обсудили несколько полезных инструментов взаимодействия.

В следующей главе мы узнаем о том, как Intruder в Burp Suite может использоваться для автоматизации сценариев атак, таких как брутфорс и т. д.

Упражнения

- Просмотрите любой из выбранных целевых URL-адресов и обратите внимание на изменения на дашборде и вкладке "Цель" (Target).
- Запустите каждый из инструментов взаимодействия с выбранной целью.

Глава 5

Intruder

В предыдущей главе мы познакомились с некоторыми основами дашборда Burp Suite, вкладкой "Цель" (Target) и инструментами взаимодействия. Теперь, когда мы ознакомились с основами перехвата запросов и интерпретации сводки в дашборде, мы перейдем к использованию инструмента Intruder. Intruder обладает расширенными возможностями фаззинга, которые могут быть использованы в различных сценариях атаки.

Введение в Intruder

Прежде чем мы углубимся в детали различных опций Intruder, важно понять, что такое Intruder и как он может быть полезен при тестировании безопасности веб-приложений. Intruder является частью Burp Suite, который может быть эффективно использован для фаззинга и выполнения атак брутфорса.

Приложение может быть со страницей входа, на которой пользователю необходимо ввести учетные данные, чтобы продолжить. С точки зрения тестирования безопасности было бы целесообразно проверить эту страницу входа на наличие учетных данных по умолчанию, слабых паролей или механизмов блокировки. Вот где Intruder может пригодиться. Имея список имен пользователей и паролей, Intruder может попробовать все эти комбинации, чтобы увидеть, подходят ли какие-либо из них.

Мы также можем рассмотреть другой сценарий, в котором у нас есть интересный запрос, который мы хотим изучить дополнительно, чтобы проверить, уязвим ли он для внедрения SQL или межсайтового скриптинга. Опять же, Intruder может помочь в этом. Мы можем просто указать Intruder на URL-адрес и параметр, который мы хотим протестировать,

S. Rahalkar, *A Complete Guide to Burp Suite*,
https://doi.org/10.1007/978-1-4842-6402-7_5

и предоставить ему список полезных нагрузок для внедрения SQL или межсайтового скриптинга. Затем он попытается внедрить все полезные нагрузки, которые мы указали в параметр, который мы хотим протестировать. Как только это будет выполнено, нам нужно будет проверить ответы, чтобы увидеть, привела ли какая-нибудь полезная нагрузка к эксплуатации уязвимости.

Таким образом, Intruder чрезвычайно помогает в любом из тестовых сценариев, где у нас есть две следующих вещи:

1. URL-адрес и параметр для тестирования
2. Список полезных нагрузок, которые должны быть отправлены в параметр

Теперь давайте попробуем понять, как мы можем отправить запрос в Intruder. Мы уже видели вкладку "Цель" (Target) и горячие клавиши в предыдущих главах. Любой запрос может быть отправлен в Intruder двумя способами:

1. Щелкните правой кнопкой мыши по запросу, который вы хотите отправить, и нажмите "Send to Intruder", как показано на рисунке 5-1.

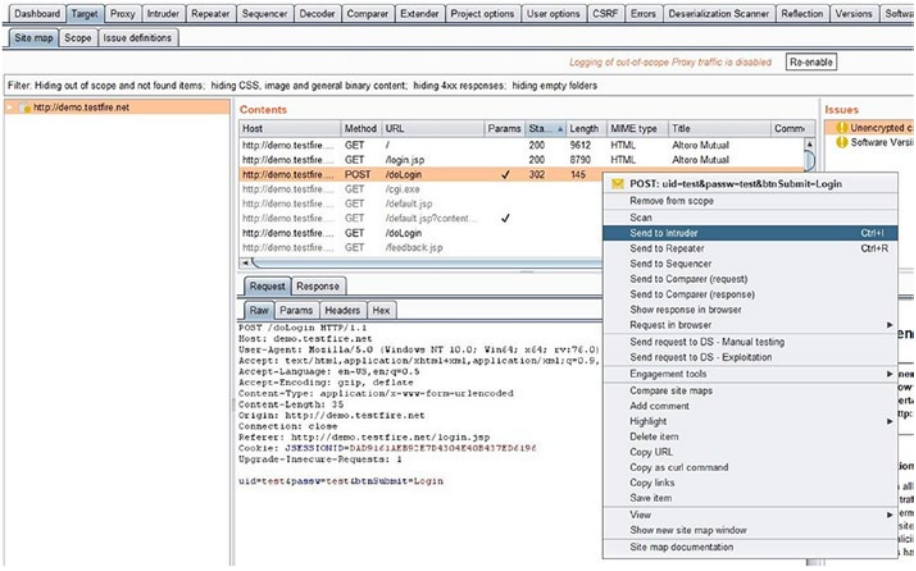

Рис.5-1. *Отправка запроса в Intruder*

2. Выберите запрос, который вы хотите отправить, и нажмите комбинацию горячих клавиш "Ctrl + I."

Теперь, когда мы отправили запрос в Intruder, давайте посмотрим, какие параметры необходимо настроить дополнительно.

Вкладка цель

Первая вкладка в Intruder - это Целевая вкладка. Здесь указан целевой URL-адрес и порт, которые мы хотим атаковать с помощью Intruder, как показано на рисунке 5-2.

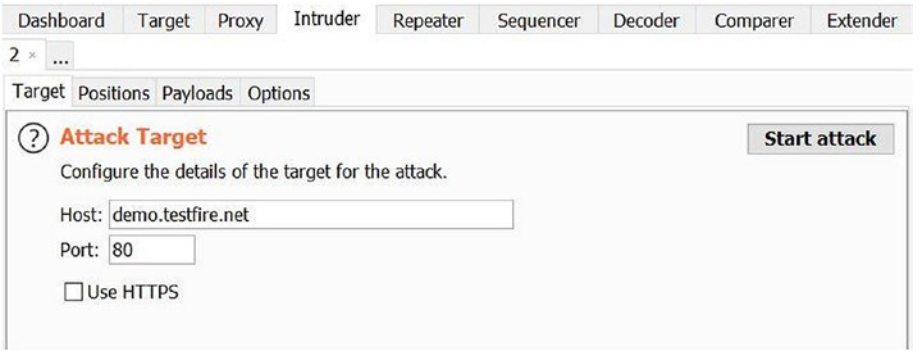

Рис.5-2. *Настройка атакуемой цели в Intruder*

Существует также возможность использовать HTTPS в случае, если целевой URL-адрес использует безопасный канал связи.

Позиции

Следующая вкладка в Intruder - это вкладка "Позиции", как показано на рис.5-3.

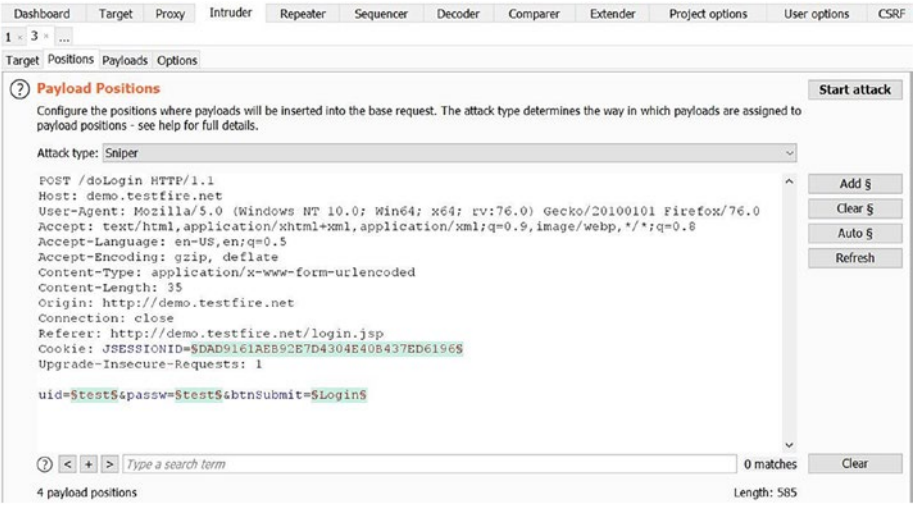

Рис.5-3. *Конфигурация позиций в Intruder*

Каждый раз, когда запрос отправляется в Intruder, он сканирует запрос на наличие возможных точек вставки и помечает их как переменные, начинающиеся и заканчивающиеся знаком "$". Существует три варианта выбора точек вставки:

1. **Add $** – Этот параметр используется для добавления новой точки вставки. Просто наведите курсор на начало и конец точки вставки и нажмите "Add $".
2. **Clear $** – Эта опция просто удалит все точки вставки, которые были выбраны вручную или автоматически.
3. **Auto $** – Эта опция будет сканировать запрос и попытается автоматически установить точки вставки, помечая их знаком "$".

Как только мы будем уверены в точках вставки или параметрах, на которые мы хотим нацелиться, следующим шагом будет выбор типа атаки. Существует четыре различных типа атак, доступных, как показано на рис. 5-4.

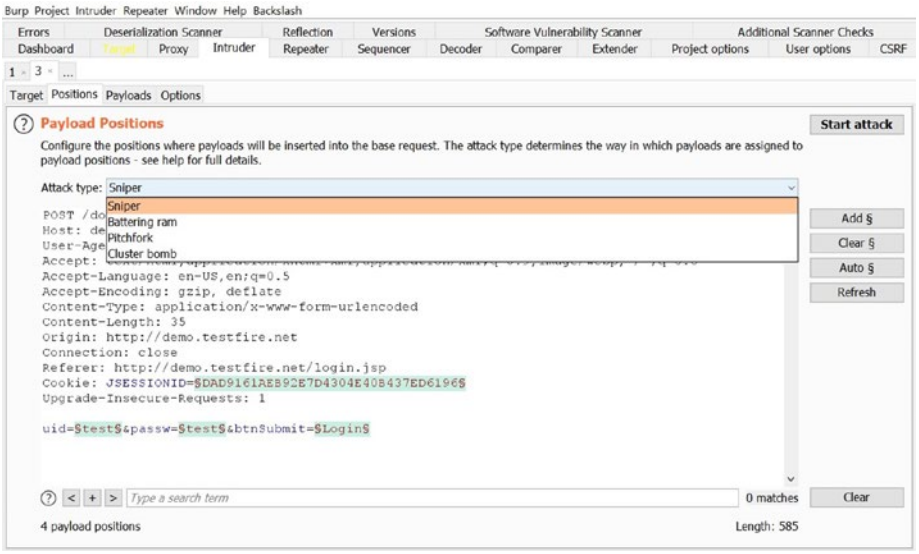

Рис.5-4. *Выбор типа атаки в Intruder*

Четыре типа атак следующие:

1. **Sniper** – Этот тип атаки использует один набор полезных нагрузок. В этом случае Intruder вставляет полезную нагрузку сразу в каждую из точек вставки, а затем перебирает ее.

2. **Battering ram** – Этот тип атаки использует один набор полезных нагрузок. В этом случае Intruder перебирает полезные нагрузки, вставляя одну и ту же полезную нагрузку во все точки вставки одновременно.

3. **Pitchfork** – Этот тип атаки использует несколько наборов полезных нагрузок. В этом случае Intruder использует различную полезную нагрузку для каждой из точек вставки.

4. **Cluster bomb** – Этот тип атаки использует несколько наборов полезных нагрузок. Для каждой из определенных точек вставки существует свой набор полезной нагрузки. Intruder перебирает каждый из наборов полезной нагрузки, и затем тестируются все перестановки комбинаций полезной нагрузки. Из-за количества возможных перестановок и комбинаций в случае cluster bomb будет сгенерировано большое количество запросов.

Выбор правильного типа атаки зависит от сценария атаки и количества точек вставки, которые необходимо атаковать одновременно. См. Рис. 5-5.

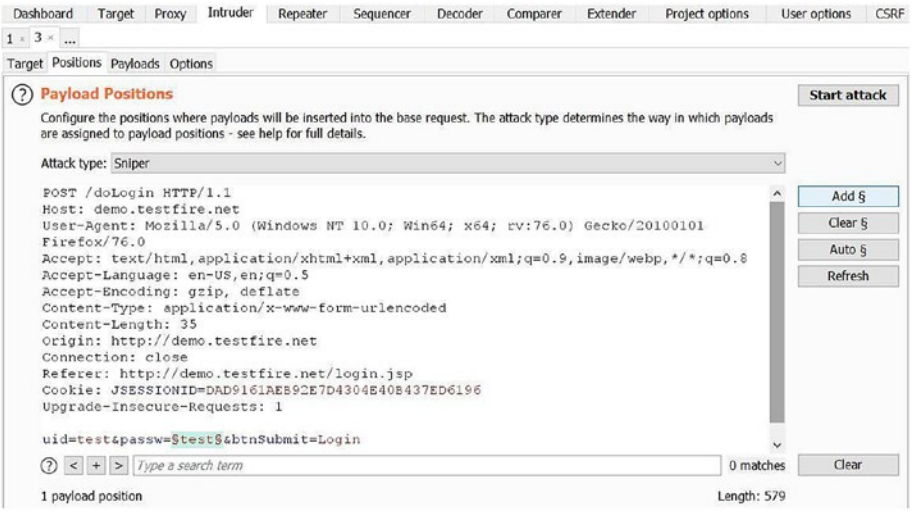

Рис.5-5. *Тип атак и позиции Intruder*

Как только позиции полезной нагрузки настроены и выбран тип атаки, мы можем перейти к настройке фактических полезных нагрузок.

Полезные нагрузки

Полезная нагрузка - это данные, которые Intruder будет итеративно вставлять в выбранные точки вставки. Полезная нагрузка может сильно отличаться в зависимости от сценария или атаки, на которую мы нацелены. В случае страницы входа, которую мы обсуждаем, полезной нагрузкой будет список возможных паролей. Burp Suite предоставляет различные типы полезной нагрузки, и наиболее распространенным из них является список. Вы можете создать свой собственный список, добавляя элементы по одному, как показано на рис.5-6, или вы также можете выбрать заранее установленный список, который предлагает Burp Suite.

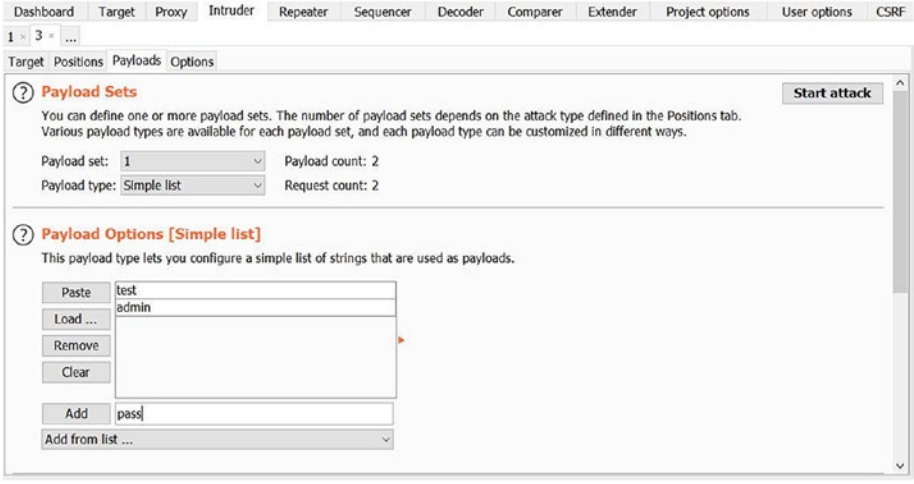

Рис.5-6. *Выбор полезных нагрузок в Intruder*

Burp Suite имеет несколько предустановленных списков в виде имен пользователей, паролей, коротких слов, полезной нагрузки для внедрения SQL и межсайтовых сценариев, директорий, расширений и т. д. В зависимости от типа атаки мы можем либо использовать предустановленный список, либо создать свой собственный, как показано на рис.5-7.

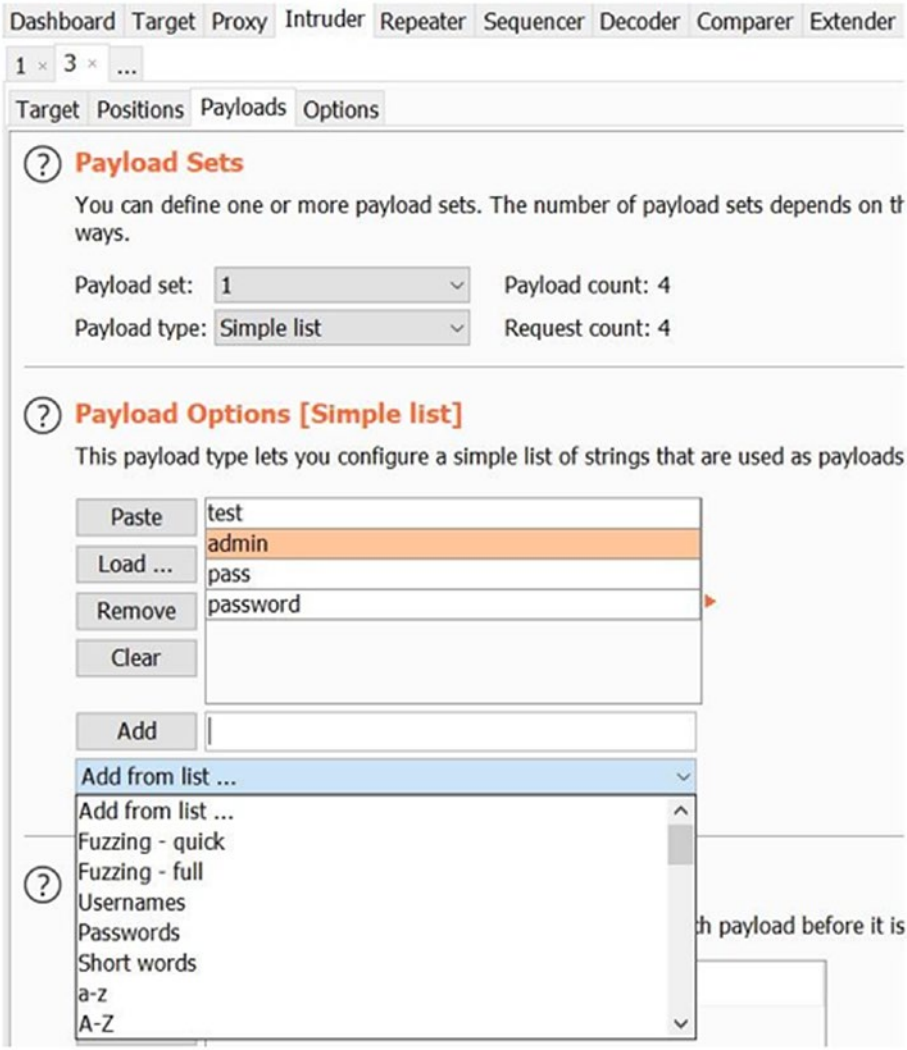

Рис.5-7. *Выбор полезной нагрузки в Intruder из различных вариантов*

Теперь, когда мы настроили позиции, а также полезную нагрузку, мы можем начать атаку, нажав кнопку "Start attack". Появится новое окно, как показано на рис.5-8, и предоставленные нами полезные нагрузки будут отправляться в точках вставки, которые мы указали ранее, – по одному запросу за раз.

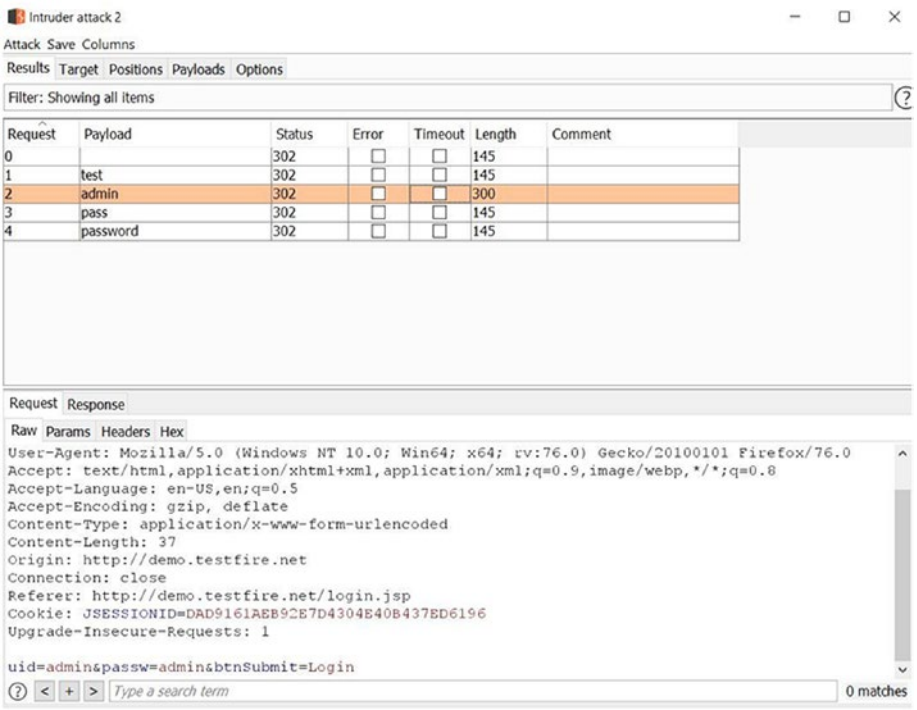

Рис.5-8. *Результаты атаки Intruder*

Из рисунка 5-8 мы видим, что Intruder отправил пять запросов, каждый с разной полезной нагрузкой. Наблюдая и сравнивая длину содержимого, мы можем заметить, что для "admin" полезной нагрузки ответ был другим. Следовательно, это может быть пароль для пользователя администратора, в который мы пытаемся войти. Затем мы можем легко проверить это вручную, войдя в целевое приложение.

Опции

Последняя часть Intruder - это вкладка "Options". Мы уже видели, что Intruder работает как инструмент фаззинга и может выполнять атаку брутфорса. Это означает, что движку Burp Suite придется отправлять большое количество запросов, ждать ответы, а затем обрабатывать их на основе предопределенного набора правил. Опция "Request Engine", как показано на рис. 5-9, помогает настроить количество параллельных потоков, количество повторных попыток при сбое в сети и паузу перед продолжением повторной попытки. Значения, показанные на рис. 5-9, заданы по умолчанию и предварительно настроены. Однако в зависимости от конкретных случаев использования эти значения могут быть соответствующим образом скорректированы.

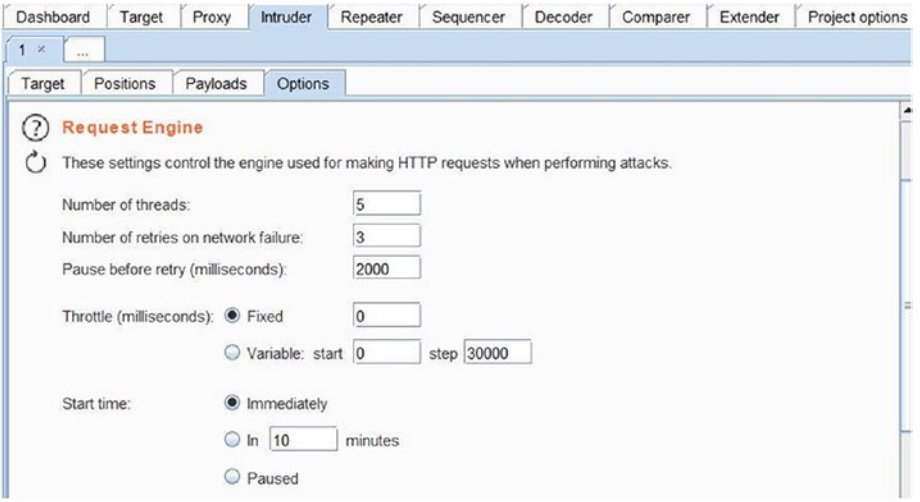

Рис.5-9. *Параметры конфигурации Intruder*

Intruder отправляет большое количество запросов к цели вместе с несколькими подстановками и комбинациями полезных нагрузок. Ответы могут быть ошеломляющими. Именно здесь пригодится функция "Grep Match", как показано на рис.5-10. С помощью этой функции мы можем настроить механизм Intruder, чтобы отмечать или выделять интересные ответы, содержащие ключевые слова,

такие как error, exception, illegal, fail, stack, access, directory и т.д. Если Intruder найдет эти ключевые слова в любом из ответов, они будут выделены, что значительно упростит анализ.

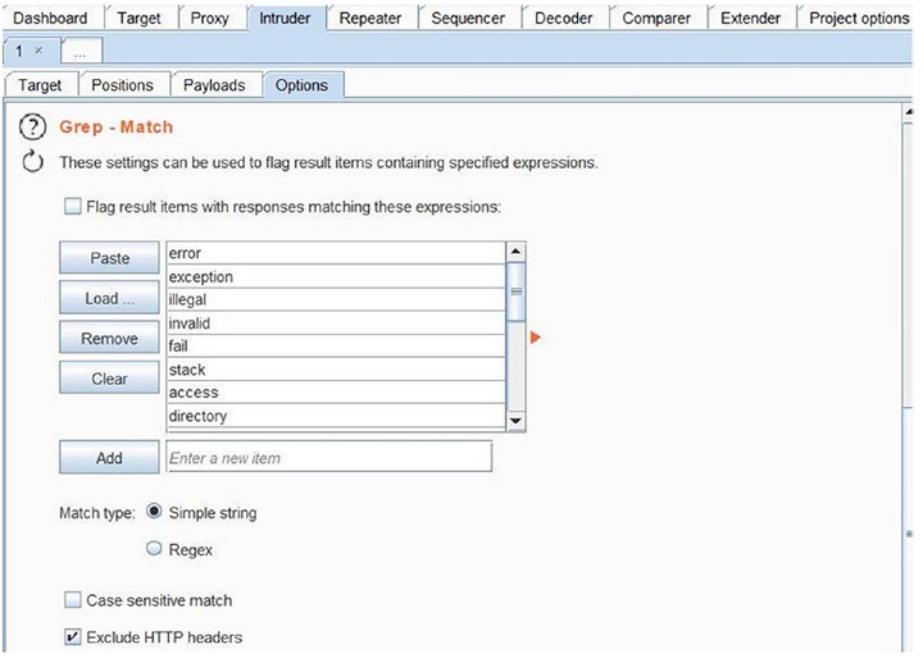

Рис.5-10. *Извлечение необходимых данных из результатов Intruder*

Резюме

В этой главе мы узнали об использовании инструмента Intruder для выполнения атак с использованием фаззинга и брутфорса. Мы начали главу с изучения того, как отправлять запросы в Intruder, настраивать позиции, полезную нагрузку и, наконец, запускать атаку и интерпретировать результаты. Мы также увидели некоторые настраиваемые параметры для Intruder.

В следующей главе мы увидим некоторые дополнительные полезные инструменты в Burp Suite, такие как Repeater, Comparer, Decoder и Sequencer.

Упражнения

1. Используйте Intruder для обнаружения уязвимости межсайтового скриптинга в любом из уязвимых приложений.
2. Используйте Intruder для обнаружения уязвимости SQL-инъекции в любом из уязвимых приложений.

Глава 6

Repeater, Comparer, Decoder и Sequencer

В предыдущей главе мы узнали о том, как Intruder можно использовать для фаззинга и выполнения атак брутфорса. В этой главе мы рассмотрим еще несколько инструментов Burp Suite, таких как Repeater, Comparer, Decoder и Sequencer.

Repeater

Repeater, как следует из названия, - это простой инструмент в Burp Suite, который помогает в воспроизведении запросов. Мы уже видели в предыдущих главах, что при просмотре приложения с помощью Burp Suite фиксируется большое количество запросов. Не все полученные запросы могут быть полезны для дальнейшего тестирования или анализа. Однако может быть несколько запросов с интересными параметрами, на которые стоит потратить время для дальнейшего анализа. Repeater помогает именно в этом сценарии. Если мы найдем конкретный запрос, заслуживающий дальнейшего изучения, мы можем просто отправить его в Repeater. Оказавшись в Repeater, мы можем поиграть с запросом так, как нам хочется; изменить его заголовки, параметры и т.д.; А затем отправить запрос приложению и посмотреть, как оно отреагирует. Repeater - это очень простой, но мощный инструмент, имеющий простой интерфейс, как показано на рис.6-1.

S. Rahalkar, *A Complete Guide to Burp Suite*,
https://doi.org/10.1007/978-1-4842-6402-7_6

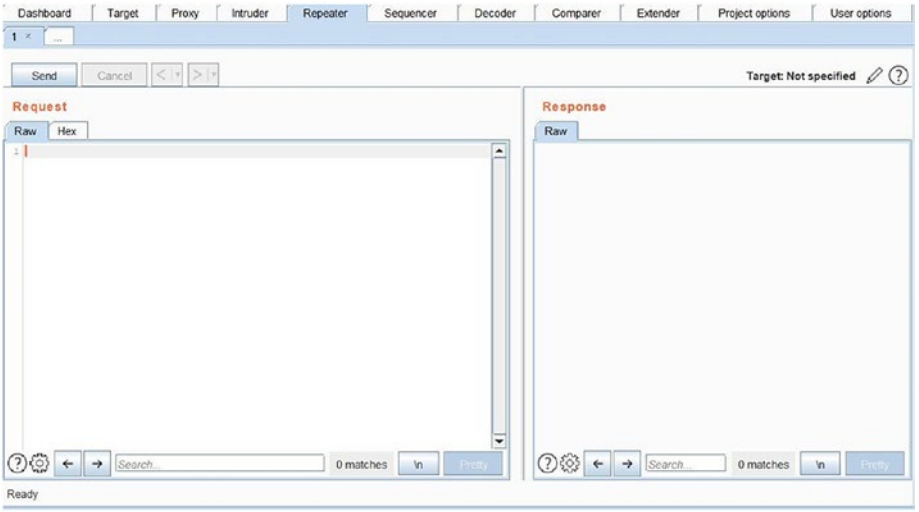

Рис.6-1. *Вкладка Repeater*

Вкладка Repeater доступна непосредственно в Burp Suite. По умолчанию он открывается пустым, и нам нужно ввести в него соответствующие данные HTTP-запроса. Чтобы указать цель, нажмите на значок "edit' рядом с "Target", и появится новое окно, как показано на рис. 6-2, в котором мы можем ввести данные хоста и порта, с которыми мы хотим взаимодействовать.

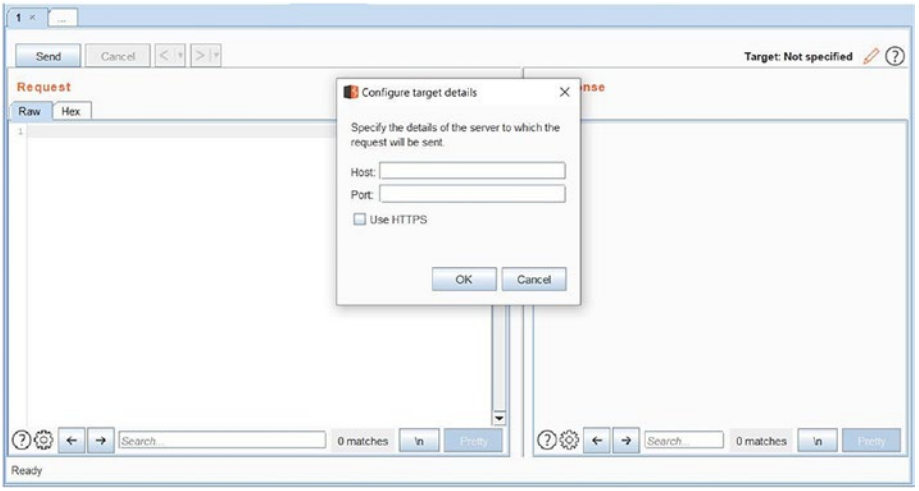

Рис.6-2. *Конфигурирование данных о цели в Repeater*

Более удобный способ отправки данных в Repeater, как показано на рисунке 6-3. Просто щелкнуть правой кнопкой мыши по любому запросу, который вы хотите отправить в Repeater для дальнейшего анализа, и нажать "Send to Repeater". Другой альтернативой является выбор запроса для отправки в Repeater и нажатие горячей клавиши "Ctrl + R.".

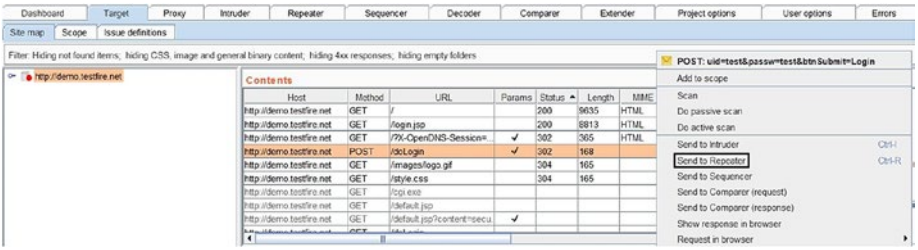

Рис.6-3. *Отправка HTTP-запроса в Repeater*

Запрос, подлежащий исследованию, теперь отправляется в Repeater, как показано на рисунке 6-4. Вкладка request содержит выбранный нами HTTP-запрос, а вкладка response пуста, так как мы еще не отправили запрос.

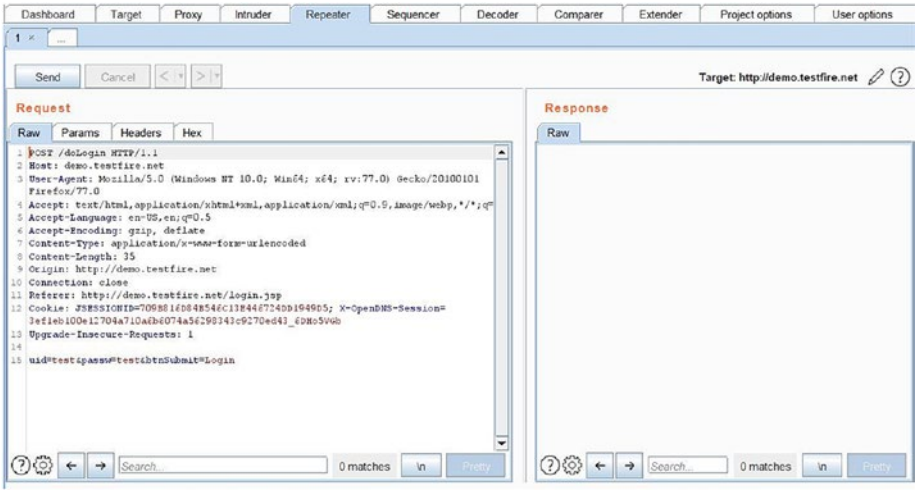

Рис.6-4. *Необработанный HTTP-запрос в Repeater*

Окно Request содержит несколько вкладок, как показано на рис. 6-5. Одна из вкладок - "Params", в которой перечислены все параметры, по умолчанию связанные с загруженным запросом. Мы можем редактировать существующие параметры, добавлять новые параметры или даже удалять любые из существующих параметров. Всегда интересно посмотреть, как приложение реагирует на все эти изменения параметров.

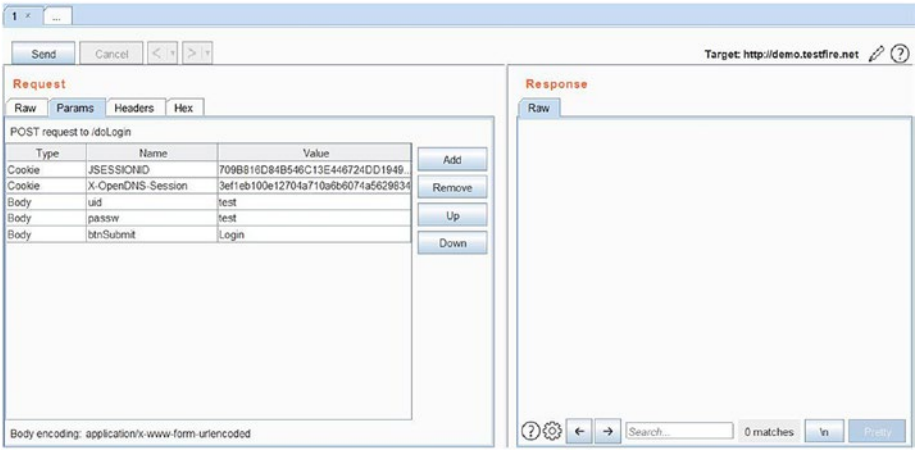

Рис.6-5. *Вкладка параметров в Repeater*

Следующая вкладка - "Headers", как показано на рис.6-6, на которой перечислены все заголовки по умолчанию, которые будут отправлены вместе с запросом. Опять же, все значения заголовка доступны для редактирования: мы можем редактировать существующие значения, добавлять новые поля заголовка, а также удалять любые существующие.

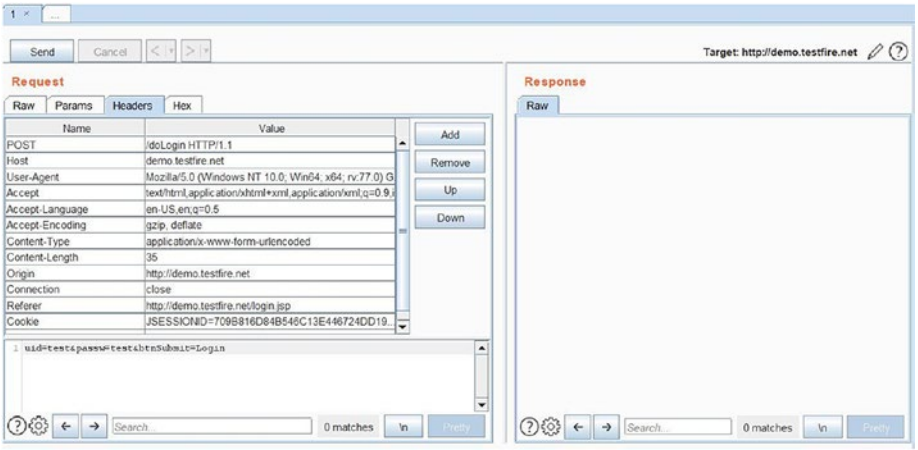

Рис.6-6. *Вкладка Заголовков в Repeater*

Как только мы установим необходимые параметры и значения заголовка, мы можем нажать кнопку "Send", как показано на рис. 6-7, и мы получим ответ от приложения, который показан на вкладке "Response".

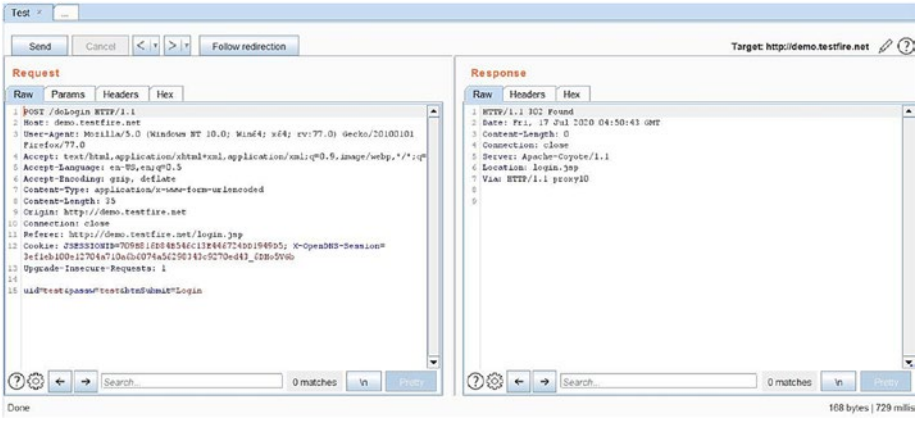

Рис.6-7. *Вкладка "Response" в Repeater*

В этом случае был получен ответ HTTP со статусом 302, что означает, что страница предназначена для перенаправления. Мы можем нажать опцию "Follow redirection", а затем получить окончательный ответ, как показано на рис.6-8.

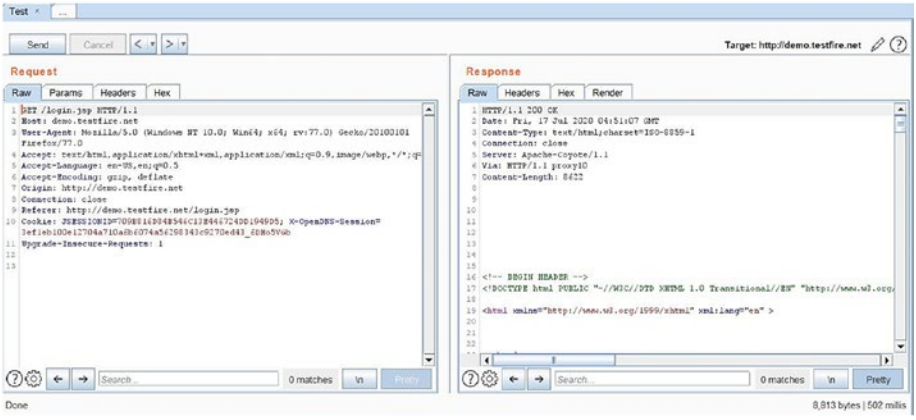

Рис.6-8. *Вкладка "Response" в Repeater*

Приведенный здесь ответ представляет собой необработанный текст, и иногда его бывает трудно интерпретировать. Следовательно, мы можем щелкнуть вкладку "Render" в разделе "Response", чтобы визуально загрузить ответ, как если бы мы видели его в браузере, как показано на рис.6-9.

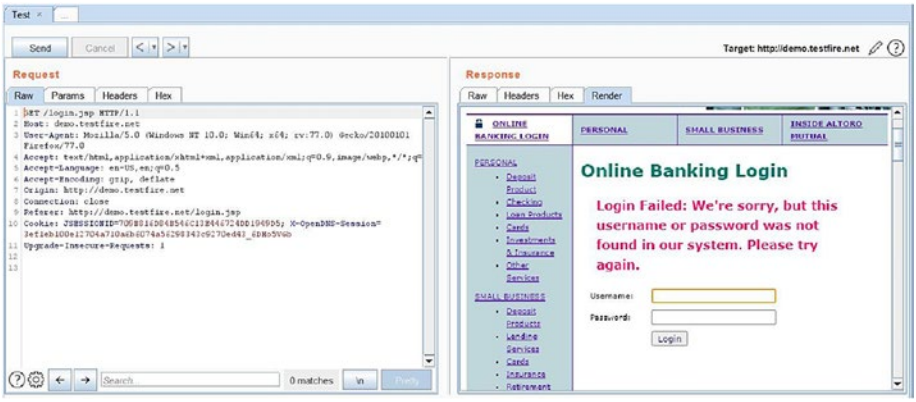

Рис.6-9. *Отображение ответа в Repeater*

Repeater также предоставляет возможность просмотра ответа в реальном браузере. Для этого просто щелкните правой кнопкой мыши на ответе, который вы хотите увидеть в браузере, и нажмите "Show response in browser", как показано на рис. 6-10.

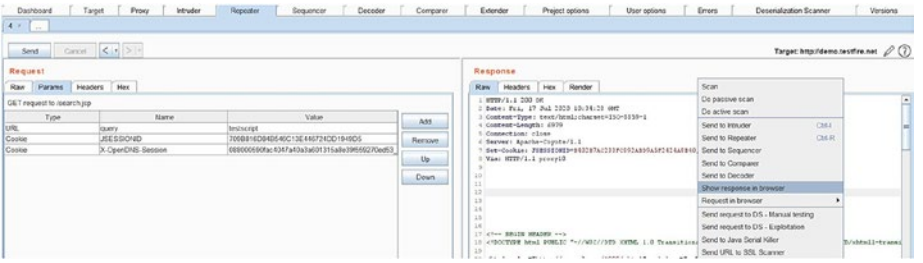

Рис.6-10. *Просмотр ответа в браузере*

Burp Suite откроет другое окно, как показано на рис. 6-11, со ссылкой, которую необходимо скопировать в браузер.

Рис.6-11. *Показать ответ в браузере*

Как только ссылка, сгенерированная Burp Suite, копируется в браузере, ответ отображается соответствующим образом.

Comparer

В предыдущем разделе мы познакомились с инструментом Repeater. Как только запрос отправляется в Repeater, существует широкая возможность изменить параметры запроса или поля заголовка и отправить его целевому приложению. Ответы в каждом конкретном случае могут отличаться в зависимости от того, какие параметры или поля заголовка были заданы в запросе. Ответов может быть несколько, и каждый из них выглядит совершенно одинаково. Вот где пригодится инструмент Comparer. Comparer просто сравнивает содержимое ответов и выделяет различия, если таковые имеются. Чтобы отправить ответ в Comparer, просто щелкните по ответу правой кнопкой мыши и нажмите "Send to Comparer", как показано на рис.6-12.

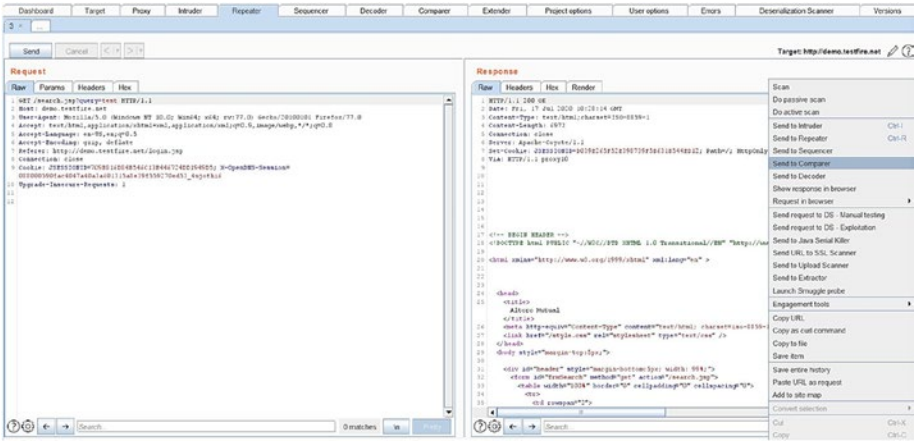

Рис.6-12. *Отправка ответа Repeater в Comparer*

Теперь, когда для сравнения Comparer требуется по крайней мере два текстовых блока, мы отправляем другой ответ Comparer, как показано на рис. 6-13.

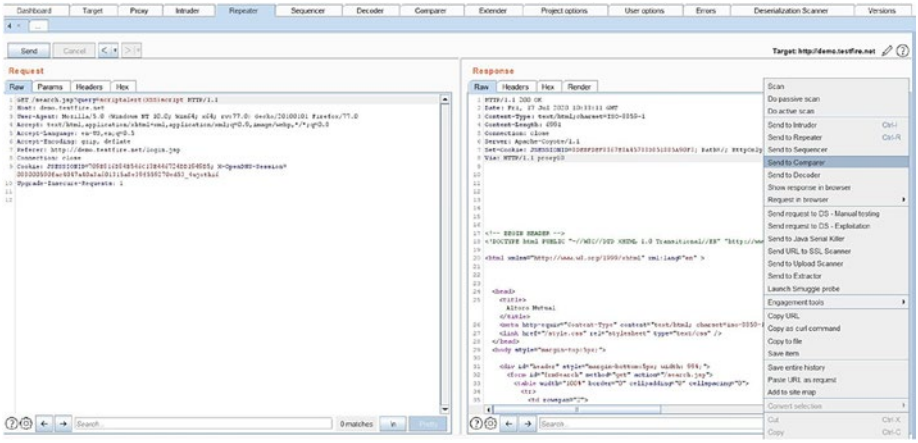

Рис.6-13. *Отправка ответа Repeater в Comparer*

Теперь мы можем перейти на вкладку "Comparer", как показано на рис.6-14, и просмотреть оба ответа, которые мы отправили с Repeater ранее.

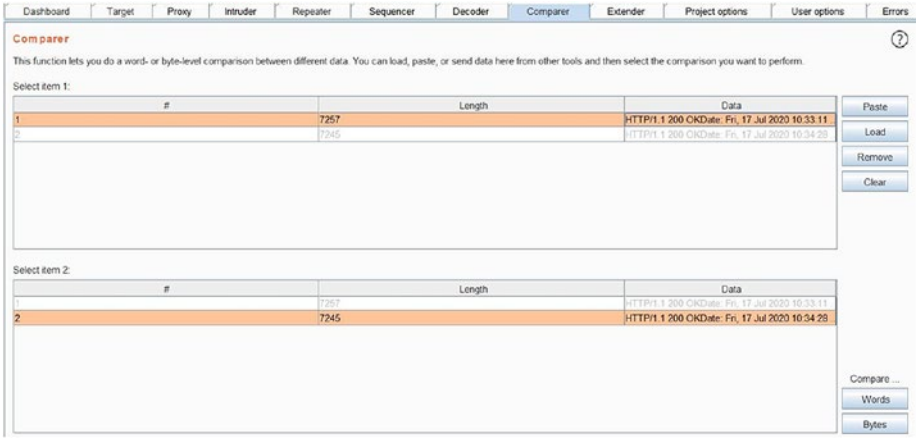

Рис.6-14. *Вкладка Comparer*

Теперь, поскольку нам нужно найти различия в словах из обоих ответов, мы нажимаем на кнопку "Words" и открываем новое окно, как показано на рис.6-15.

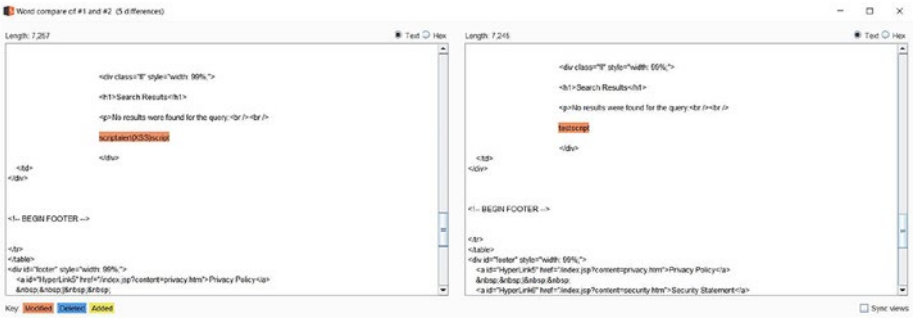

Рис.6-15. *Сравнение ответов в Comparer*

Теперь Comparer выделит изменения текста в двух параллельных окнах.

Decoder

Веб-приложения обычно используют различные схемы кодирования, такие как Ascii, HTML, Base 64 и т.д. С точки зрения тестирования безопасности, очень часто во время тестирования встречаются такие закодированные строки. Burp Suite Decoder - это простая утилита, которая может кодировать или декодировать текст в формате URL, HTML, Base 64, ASCII hHx, Шестнадцатеричный, восьмеричный, двоичный и Gzip. Просто перейдите на вкладку Decoder, как показано на рис.6-16, и введите текст, который необходимо декодировать.

Рис.6-16. *Декодирование Base 64 данных*

В этом случае мы ввели кодированное значение Base 64, а затем нажали "Decode as" Base 64, чтобы получить декодированный вывод в следующем окне как admin:admin.

Мы также можем использовать этот инструмент для кодирования любого обычного текста, как показано на рис. 6-17.

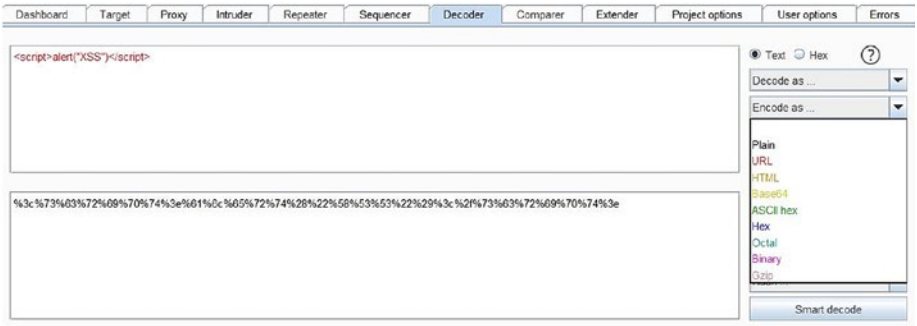

Рис.6-17. *Кодирование данных с помощью Decoder в Burp Suite*

Мы просто ввели текстовое предупреждение <script>alert(“XSS”)</script>, а затем нажали "Encode as" и URL, чтобы получить закодированный вывод в следующем окне.

Sequencer

Веб-приложения во многом зависят от токенов, идентификаторов сеансов или других подобных уникальных и случайных идентификаторов. С точки зрения безопасности важно проверить случайность или уникальность этих токенов и идентификаторов. Если токены недостаточно сильны и случайны, злоумышленники могут легко применить к ним брутфорс и получить несанкционированный доступ.

Burp Suite Sequencer - это инструмент, который помогает нам проверить надежность токенов приложений. Мы можем отправить любой запрос в Sequencer, просто щелкнув правой кнопкой мыши по запросу и нажав "Send to Sequencer", как показано на рис.6-18.

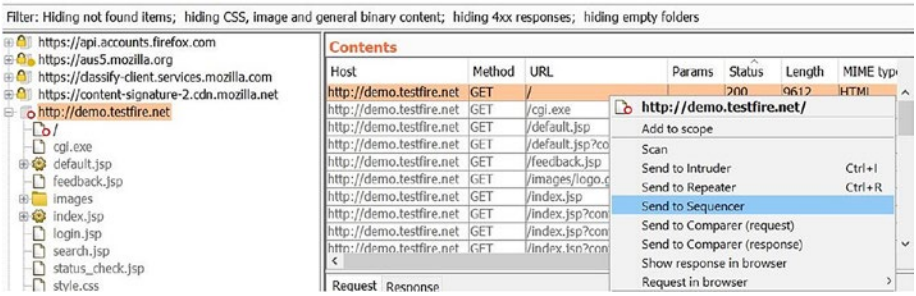

Рис.6-18. *Отправка HTTP-запроса в Sequencer*

Теперь мы можем перейти на вкладку Sequencer, как показано на рис.6-19.

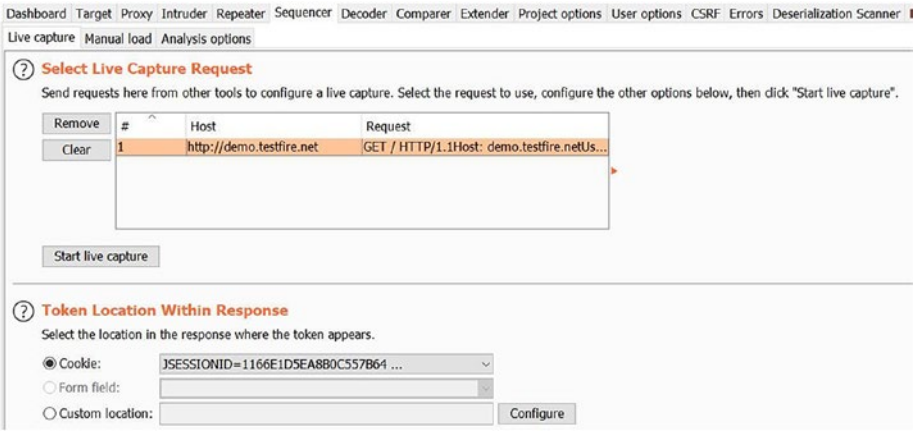

Рис.6-19. *Sequencer в Burp Suite*

Мы ясно видим, что Sequencer автоматически проанализировал запрос и выбрал токен JSESSIONID, присутствующий в файле cookie. Теперь этот токен будет проанализирован на предмет его силы и случайности. Чтобы начать тест, просто нажмите "Start live capture", и откроется новое окно, как показано на рис.6-20.

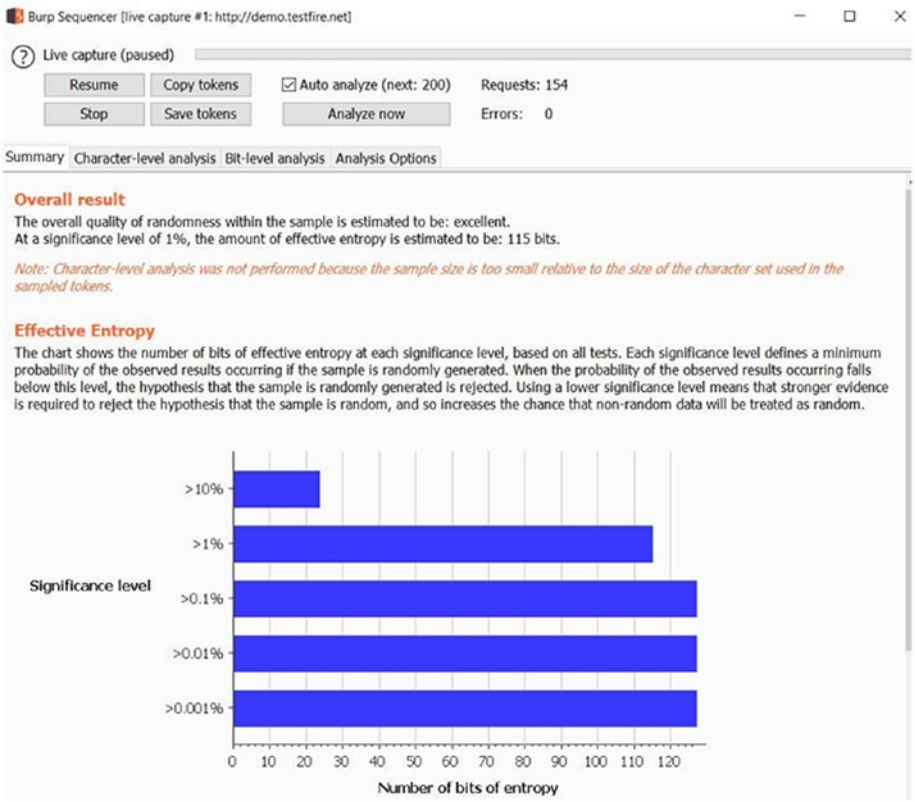

Рис.6-20. *Анализ идентификатора сеанса с использованием Sequencer*

Начнется запись в реальном времени, и мы можем приостановить или возобновить ее в любое время, когда пожелаем. Однако важно отметить, что для эффективного анализа надежности токенов следует учитывать размер выборки не менее 100. Как только захват завершен, Sequencer показывает нам результат, который в данном случае был признан “отличным”. Таким образом, токен JSESSIONID является сильным, уникальным, случайным и, следовательно, безопасным в использовании.

Sequencer также позволяет нам вручную загружать выборку токенов, а затем анализировать их. Для этого просто перейдите на вкладку "Manual load" в Sequencer, как показано на рис. 6-21.

Рис.6-21. *Ручная загрузка токенов для анализа*

Теперь мы можем скопировать и вставить образец токенов, а затем проанализировать их соответствующим образом.

Резюме

В этой главе мы рассмотрели, как подделывать и воспроизводить запросы с помощью Repeater, а затем выполнять прямое сравнение ответов с помощью Comparer. Затем мы узнали об инструменте Decoder, который помогает кодировать и декодировать текст в различных форматах. Наконец, мы познакомились с инструментом Sequencer, который можно использовать для оценки эффективности токенов.

В следующей главе мы узнаем о некоторых дополнительных полезных инструментах в Burp Suite, таких как Infiltrator, Collaborator, Clickbandit и CSRF PoC Генератор.

Упражнения

1. Отправьте любой запрос в Repeater и попытайтесь изменить его параметры и поля заголовка. Затем отправьте каждый запрос и проанализируйте ответы.
2. Сравните несколько ответов с помощью Comparer.
3. Используйте инструмент Decoder для кодирования текста в форматах URL, HTML и Base 64.
4. Оцените силу токена по любому из захваченных запросов.

Глава 7

Infiltrator, Collaborator, Clickbandit и CSRF PoC Generator

В предыдущей главе мы рассмотрели некоторые инструменты Burp Suite, такие как Repeater, Sequencer, Decoder и Comparer. В этой главе мы продолжим изучение более полезных инструментов, таких как Infiltrator, Collaborator, Clickbandit и CSRF PoC (proof-of-concept) генератор.

Infiltrator

Burp Suite Infiltrator - это инструмент, который настраивает целевое веб-приложение таким образом, чтобы обнаружение уязвимостей сканером Burp Suite становилось более эффективным и точным. Infiltrator вносит необратимые изменения в код и, по сути, подключается к целевому приложению. Таким образом, это помогает сканеру Burp Suite лучше видеть код приложения и потенциально обнаруживать небезопасные вызовы и функции.

S. Rahalkar, *A Complete Guide to Burp Suite*,
https://doi.org/10.1007/978-1-4842-6402-7_7

Поскольку Infiltrator вносит необратимые изменения в код целевого приложения, рекомендуется запускать его только для тестового экземпляра, а не для рабочего. В настоящее время Infiltrator поддерживается, если целевое приложение использует любую из следующих технологий:

- Java
- Groovy
- Scala
- Other JVM language (JRE versions 1.4 - 1.8)
- C#
- Visual Basic
- Другие .Net языки (.Net версии выше 2.0)

Чтобы начать работу с Infiltrator, нажмите на меню "Burp", а затем выберите "Burp Infiltrator", как показано на рисунке 7-1.

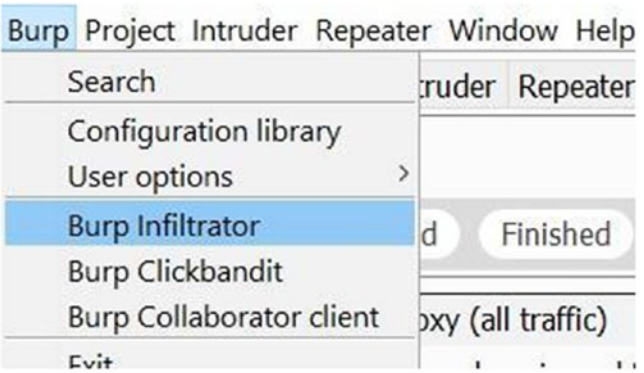

Рис.7-1. *Переход к Infiltrator Burp Suite*

Появится новое окно, как показано на рисунке 7-2. Этот мастер поможет нам создать агента Infiltrator .

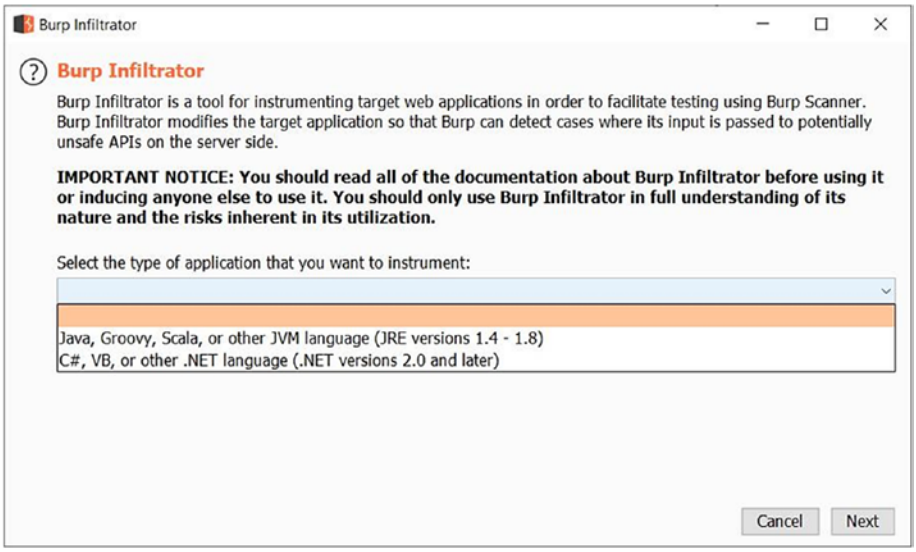

Рис.7-2. *Создание агента Infiltrator*

Нам нужно выбрать технологию, которую использует наше приложение, например Java или .NET, и нажать "Next". Затем мастер запросит местоположение, в котором мы хотим сохранить агента Infiltrator, как показано на рис. 7-3.

Рис.7-3. *Создание агента Infiltrator*

Затем мастер просто сгенерирует агента Infiltrator и сохранит его в выбранном нами ранее месте, как показано на рис. 7-4.

Рис.7-4. *Создание агента Infiltrator*

Здесь важно отметить, что агент Infiltrator должен находиться в том же каталоге, где находится целевое приложение, как показано на рис.7-5.

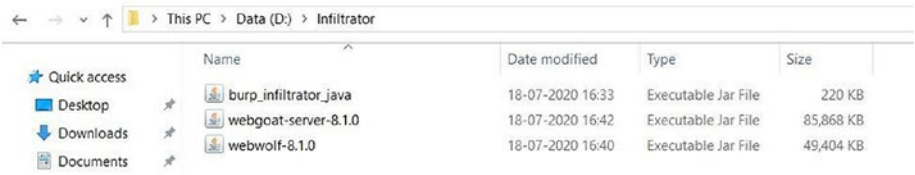

Рис.7-5. *Недавно созданный агент Infiltrator*

Теперь, когда и агент Infiltrator, и целевое приложение находятся в одном каталоге, мы можем открыть командную строку и ввести команду "java -jar burp_ infiltrator.jar' как показано на рисунке 7-6.

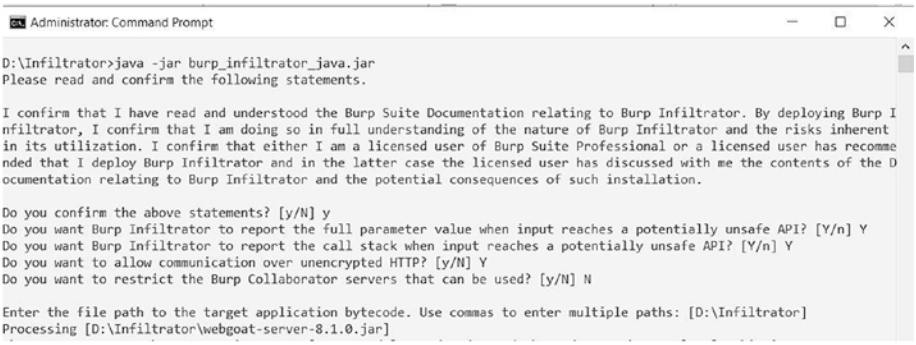

Рис.7-6. *Выполнение агента Infiltrator*

Теперь Infiltrator будет запускать и изменять приложения Java в каталоге. Это одноразовая процедура, и приложение необходимо перезапустить, как только будет доступен исправленный код. Burp Infiltrator также использует Collaborator, который мы увидим в следующем разделе.

Collaborator

Collaborator - это инструмент, предоставляемый Burp Suite, который помогает в таких атаках, как подделка запросов на стороне сервера (SSRF) или любые внеполосные атаки. Служба совместной работы Burp Suite помогает генерировать случайные полезные нагрузки в виде имен хостов. Эти полезные нагрузки затем могут использоваться как часть запросов в различных сценариях атаки. Если атака успешна, то происходит взаимодействие между целевым сервером приложения и сервером Burp Collaborator. Затем, используя клиент Burp Collaborator, мы можем узнать и проверить, происходили ли какие-либо подобные взаимодействия.

Чтобы начать работу с Burp Collaborator, просто нажмите на вкладку Burp и нажмите “Burp Collaborator client". Появится новое окно, как показано на рис. 7-7.

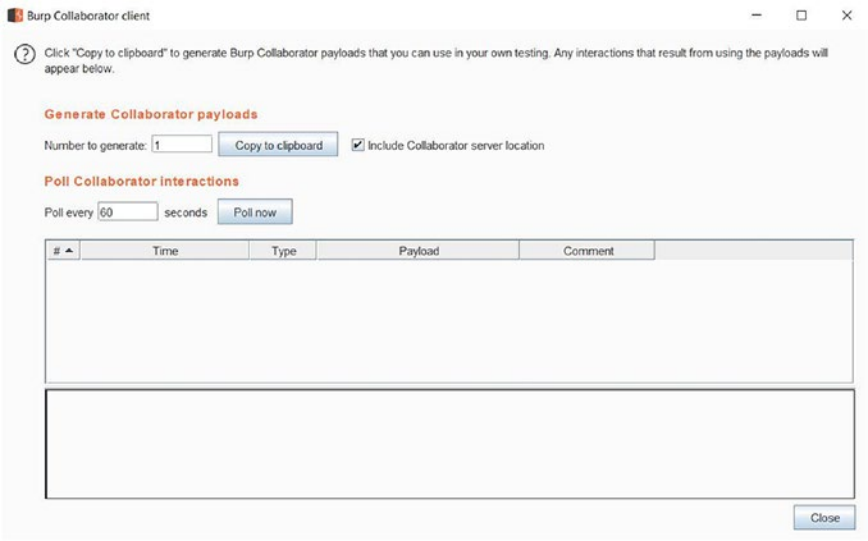

Рис.7-7. *Клиент Burp Suite Collaborator*

Теперь нажмите на кнопку "Copy to clipboard" и вставьте ее значение в блокнот, как показано на рис.7-8.

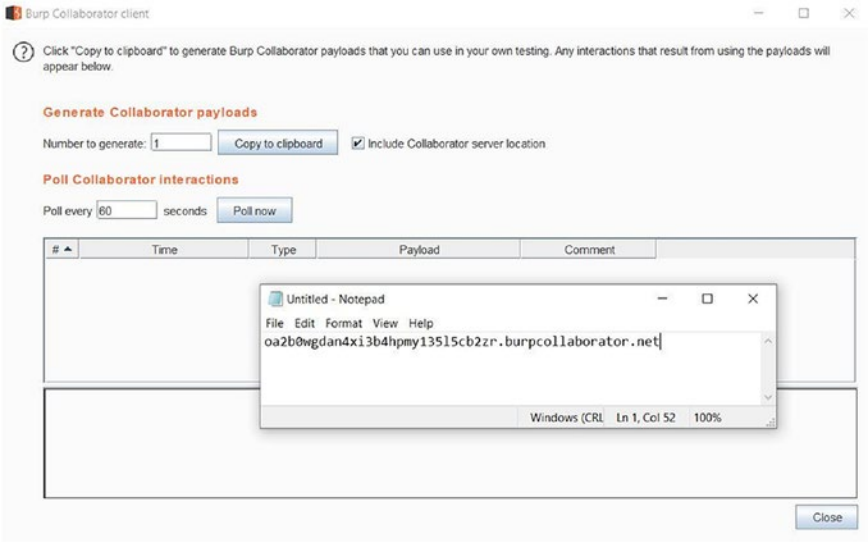

Рис.7-8. *Настройка клиента Collaborator*

Случайное значение, сгенерированное Burp Collaborator, теперь может использоваться в полезных нагрузках в запросах, отправляемых в рамках атаки. Клиент Burp Collaborator автоматически опрашивает сервер Collaborator каждые 60 секунд, чтобы проверить, было ли какое-либо взаимодействие. Эту продолжительность можно настроить, или вы можете просто нажать кнопку "Poll now", чтобы вручную проверить взаимодействие с Collaborator.

Clickbandit

Clickjacking - одна из очень распространенных атак на веб-приложения. Используя clickjacking, злоумышленник пытается обманом заставить пользователя щелкнуть что-то другое, чем то, что пользователь видит визуально. В случае успеха злоумышленник может получить доступ к конфиденциальной информации. Clickjacking также известен как атака на изменение пользовательского интерфейса, поскольку злоумышленник использует обманчивый метод создания поддельного пользовательского интерфейса, а затем обманывает жертву, заставляя ее выполнять вредоносные действия.

Burp Suite предлагает утилиту под названием "Clickbandit", которая значительно упрощает процесс создания доказательства концепции (PoC) для приложения, уязвимого для Clickjacking.
Чтобы начать работу с инструментом Clickbandit, просто перейдите во вкладку "Burp" и нажмите "Burp Clickbandit". Появится новое окно, как показано на рис. 7-9.

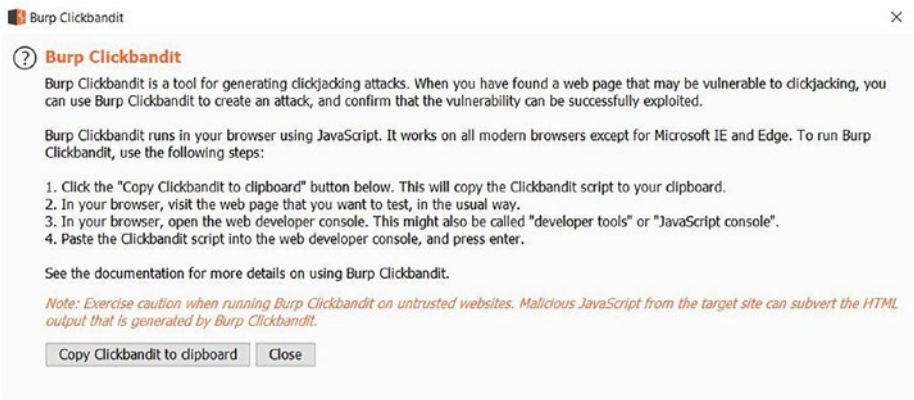

Рис.7-9. *Инструмент Clickbandit в Burp Suite*

В этом окне перечислены шаги, которые нам необходимо выполнить, чтобы создать доказательство концепции (PoC) Clickjacking. Первый шаг - нажать на кнопку "Copy Clickbandit to clipboard". Следующий шаг - открыть браузер и нажать функциональную клавишу F12, чтобы перейти в консоль браузера, как показано на рис.7-10.

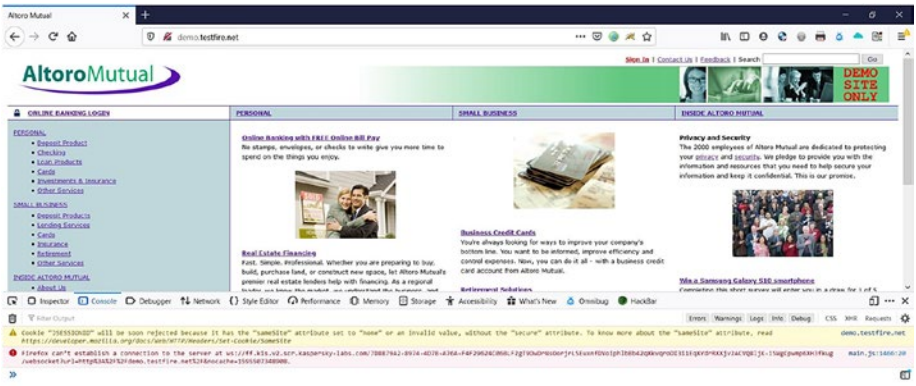

Рис.7-10. *Цель для Clickbandit*

Чтобы продолжить, нам нужно вставить код Clickbandit в эту консоль браузера, который мы скопировали ранее, как показано на рис. 7-11.

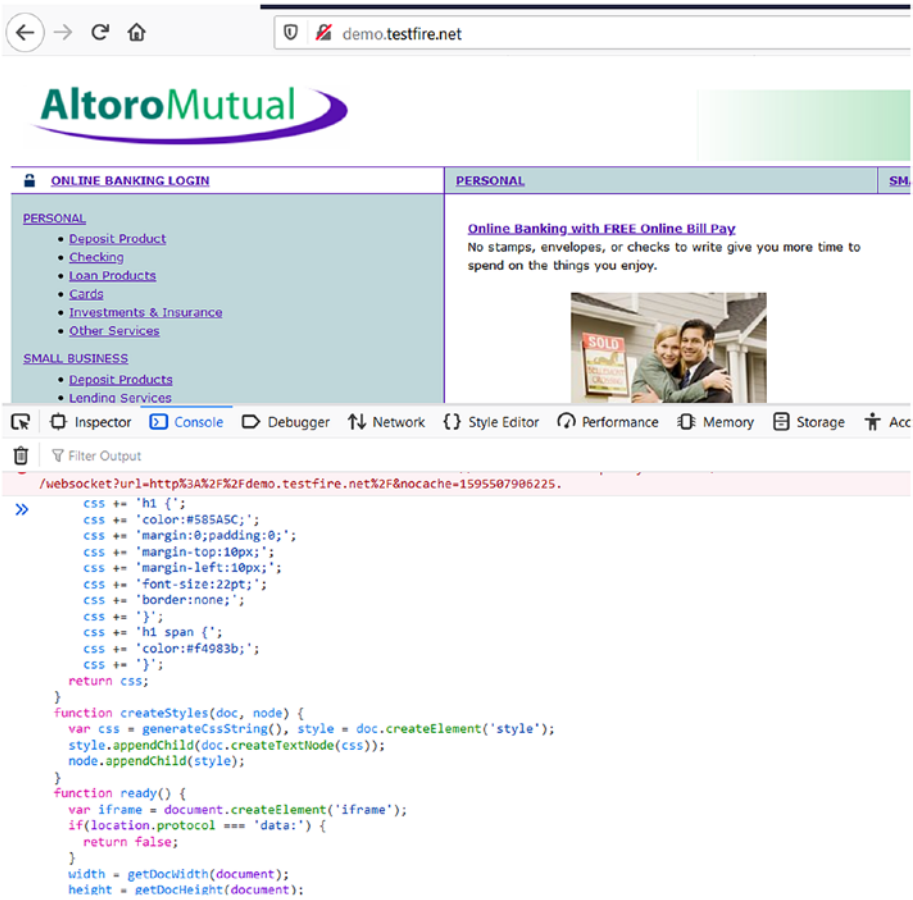

Рис.7-11. *Копирование кода Clickbandit в консоль браузера*

Как только код будет скопирован в консоль браузера, просто нажмите Enter, и вы заметите, что пользовательский интерфейс Burp Clickbandit появится в верхней части страницы, как показано на рис.7-12.

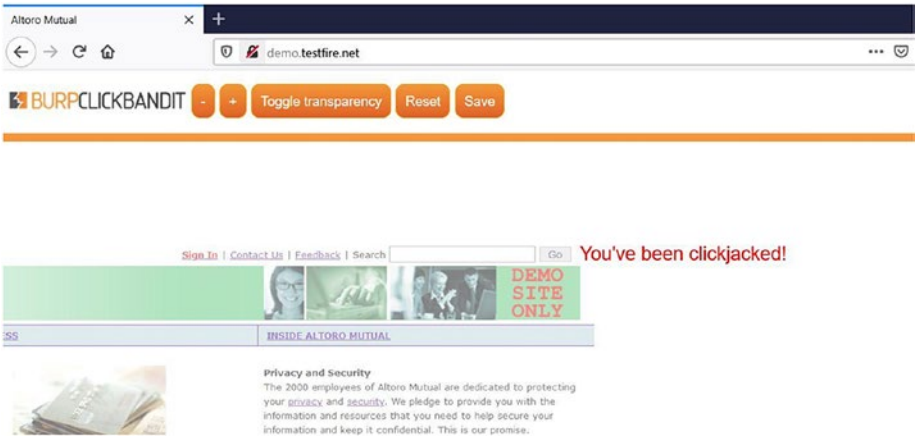

Рис.7-12. *Пользовательский интерфейс Clickbandit*

Теперь нам нужно выполнить и записать действия, которые мы хотим включить в атаку с помощью Clickjacking. Как только все необходимые действия будут выполнены, нажмите на кнопку save, и вы сможете сохранить файл с именем ‘clickjacked.html’ как показано на рисунке 7-13.

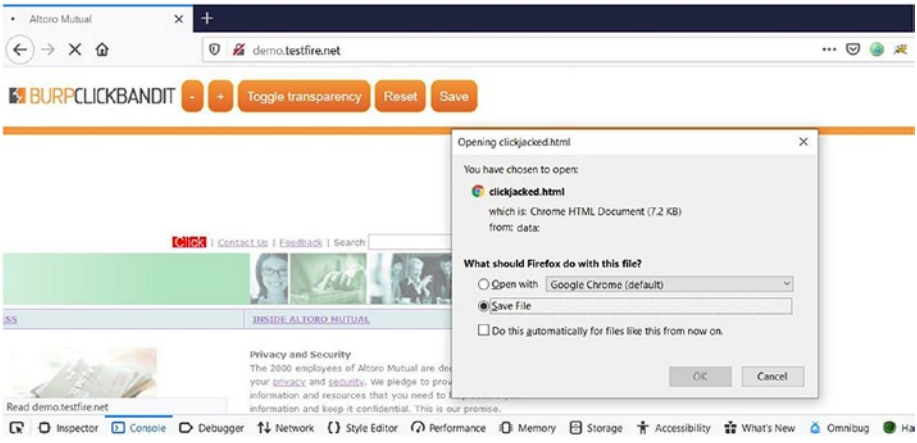

Рис.7-13. *Сохранение кода Clickbandit*

Теперь вы можете открыть файл ‘clickjacked.html’ отдельно в браузере, как показано на рис.7-14.

Рис.7-14. *Выполнение кода Clickbandit*

Вы заметите, что действия, которые вы зафиксировали ранее, теперь воспроизводятся, и если вы нажмете, то получите сообщение "You’ve been clickjacked!", как показано на рис.7-15.

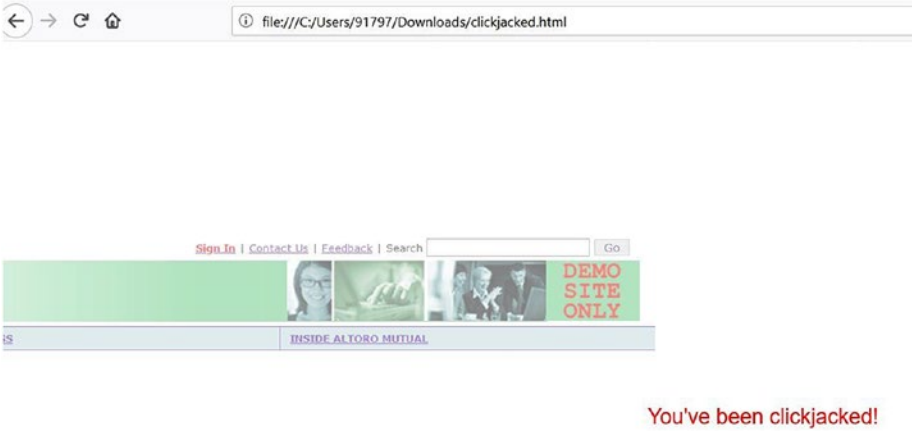

Рис.7-15. *Выполнение кода Clickbandit*

CSRF

Подделка межсайтовых запросов, широко известная как CSRF - это еще один тип атаки на веб-приложения, который использует недостатки управления сеансами, чтобы обмануть жертву и заставить ее выполнять нежелательные действия. В Burp Suite есть утилита, которая позволяет очень легко генерировать доказательство концепции (PoC) уязвимости CSRF.

Сначала нам нужно определить и подтвердить запрос, для которого мы хотим сгенерировать PoC код для CSRF. Как только запрос будет завершен, просто щелкните правой кнопкой мыши, перейдите в раздел "Engagement tools" и нажмите "Generate CSRF PoC", как показано на рис.7-16.

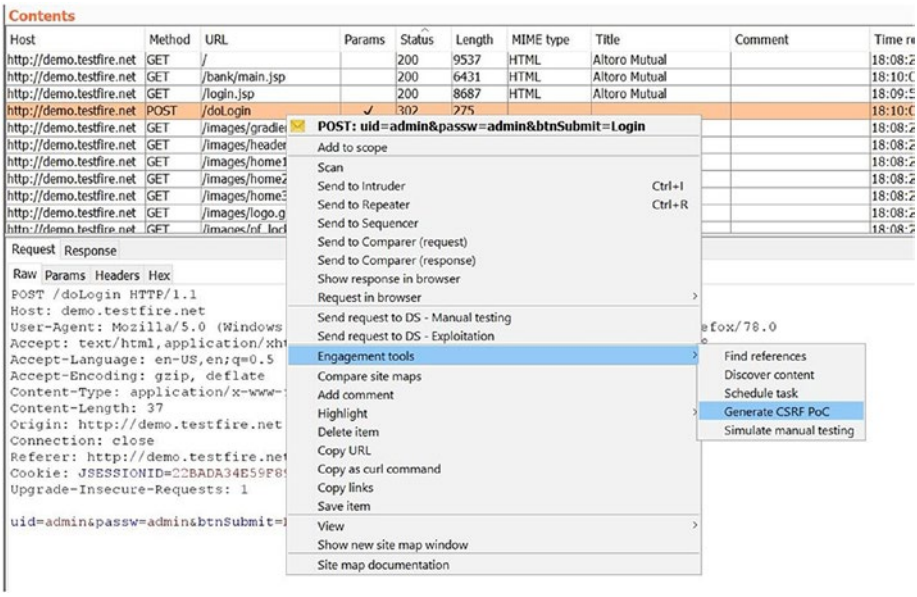

Рис.7-16. *Отправка запроса в CSRF PoC generator*

Теперь появится новое окно, как показано на рис. 7-17, в котором есть запрос POST вместе с кодом CSRF.

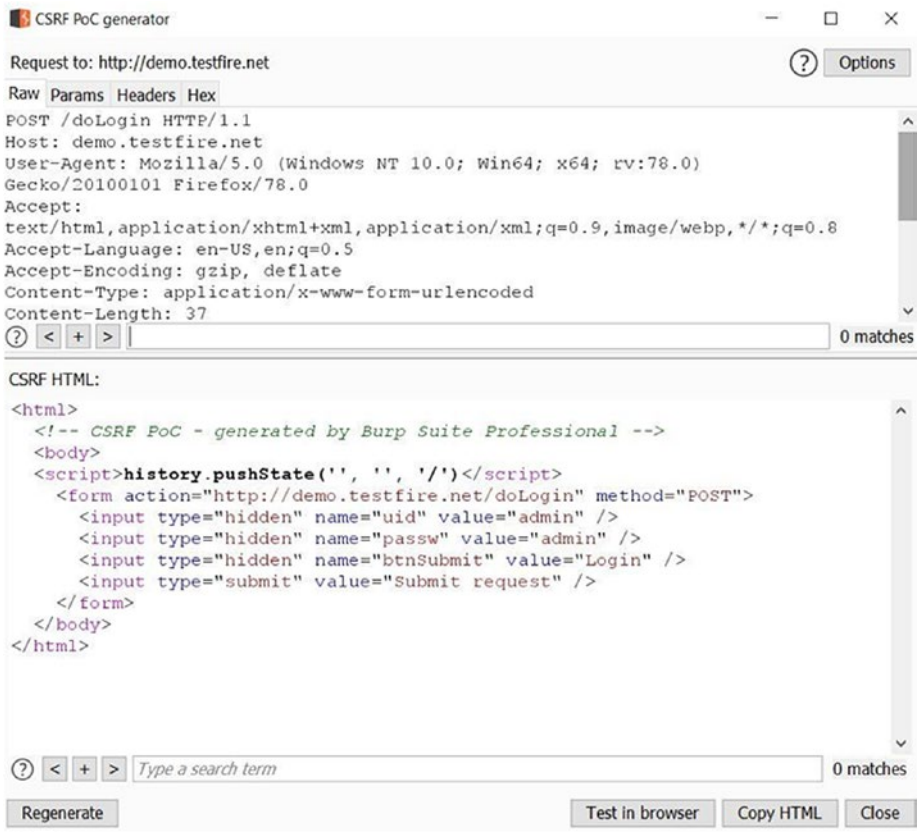

Рис.7-17. *CSRF PoC generator*

Теперь можно легко изменить код CSRF по мере необходимости, а затем мы можем либо напрямую протестировать его в браузере, либо создать отдельный HTML-файл. Чтобы проверить код CSRF в браузере, нажмите кнопку "Test in browser", и появится новое окно, как показано на рис. 7-18.

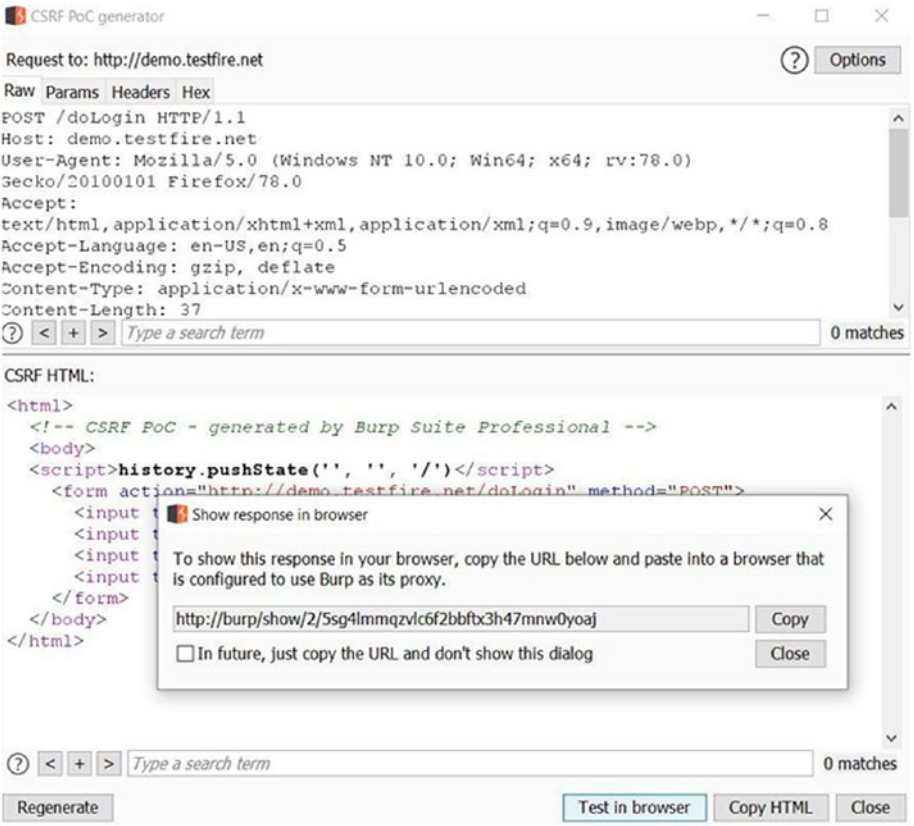

Рис.7-18. *CSRF PoC generator*

Теперь нажмите на кнопку "Copy", откройте браузер и вставьте в адресную строку, как показано на рис. 7-19.

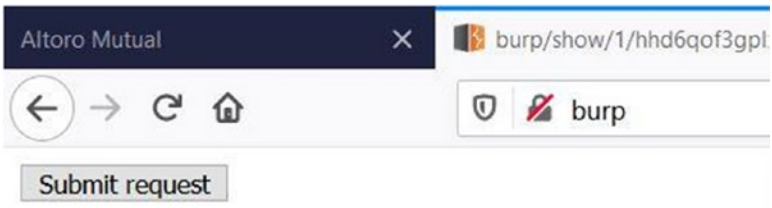

Рис.7-19. *Проверка PoC CSRF в браузере*

Теперь нажмите на кнопку "Submit request", и код CSRF будет выполнен, как показано на рисунке 7-20.

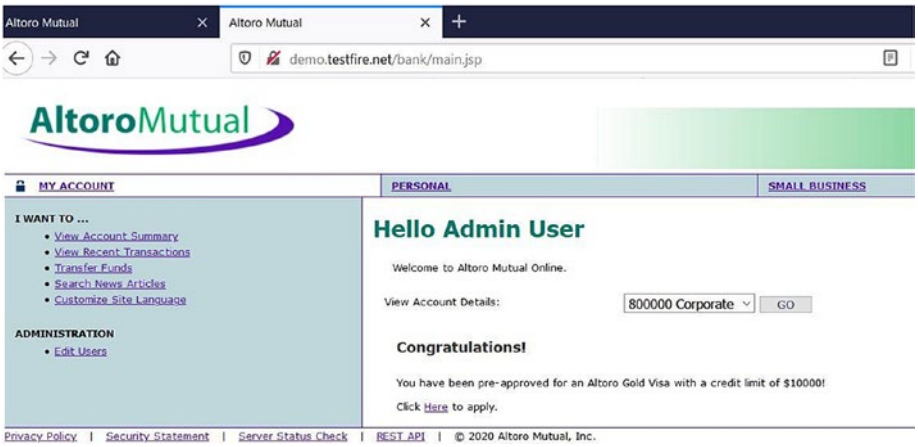

Рис.7-20. *Проверка PoC CSRF в браузере*

Резюме

В этой главе мы узнали об использовании Intruder для инструментирования приложений и расширения возможностей обнаружения Burp Scanner. Затем мы увидели Burp Collaborator, который может быть эффективно использован во внеполосных атаках, таких как SSRF. Затем мы рассмотрели инструмент Clickbandit, который помогает генерировать PoC код для приложений, уязвимых для clickjacking; и, наконец, мы просмотрели CSRF PoC generator, который помогает нам быстро генерировать и тестировать PoC код для атак на подделку межсайтовых запросов (CSRF).

В следующей главе мы рассмотрим возможности автоматического сканирования и отчетности Burp Suite.

Упражнения

1. Используйте Infiltrator для проверки любого из ваших целевых приложений Java.
2. Найдите уязвимый запрос CSRF и попробуйте сгенерировать PoC код с помощью CSRF PoC generator.
3. Сгенерируйте PoC код clickjacking для вашего целевого веб-приложения.

Глава 8

Сканер и отчеты

В предыдущей главе мы узнали о различных инструментах, таких как Infiltrator, Collaborator, Clickbandit и CSRF PoC generator. В этой главе мы рассмотрим функции и возможности сканера Burp Suite для автоматического обнаружения уязвимостей.

Типы сканирования

На протяжении всей книги мы видели несколько возможностей Burp Suite, которые полезны для ручного тестирования. Однако Burp Suite также предоставляет сканер уязвимостей веб-приложений, который автоматизирует процесс поиска уязвимостей. Это действительно очень многофункциональный сканер, способный обнаруживать потенциальные веб-уязвимости.

Burp Suite предлагает два типа сканирования: Пассивное сканирование и Активное сканирование. Пассивное сканирование по умолчанию выполняется в фоновом режиме, пока мы просматриваем приложение через Burp Suite. Пассивное сканирование просто отслеживает трафик и пытается выявить уязвимости, такие как отсутствующие флаги безопасности в файлах cookie, отсутствующие заголовки безопасности, трафик, отправляемый по незашифрованным каналам связи и т.д. Таким образом, пассивный сканер не пытается вводить какие-либо полезные нагрузки в какие-либо точки вставки, а скорее просто выявляет уязвимости, которые можно обнаружить только путем пассивного мониторинга текущих запросов и ответов.

S. Rahalkar, *A Complete Guide to Burp Suite*,
https://doi.org/10.1007/978-1-4842-6402-7_8

Активное сканирование идет дальше и пытается вставить полезную нагрузку в точки вставки и проверить, являются ли параметры уязвимыми. Активное сканирование - более интенсивный метод, однако он лучше справляется с поиском уязвимостей, которых пассивный сканер может никогда не обнаружить. Теперь мы подробнее рассмотрим детали выполнения активного сканирования с помощью Burp Suite.

Обход и аудит

Активное сканирование обычно состоит из двух этапов. Первый шаг включает в себя обход или создание пауков в приложении, а второй шаг включает атаку на параметры с помощью полезных нагрузок. Сканер Burp Suite предлагает два варианта: либо обход и аудит, либо просто обход. Чтобы начать новый аудит, перейдите на вкладку "Dashboard" и нажмите "New scan", как показано на рис.8-1.

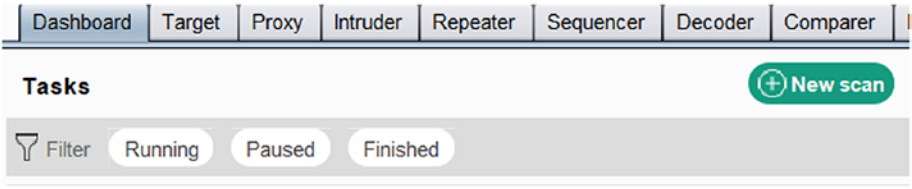

Рис.8-1. *Новая задача сканирования*

Появится новое окно, как показано на рисунке 8-2. Мы выбираем опцию "Crawl and audit". Затем нам нужно указать целевой URL-адрес, который мы хотим сканировать. В этом случае мы вводим целевой URL-адрес как ‘demo.testfire.net.’

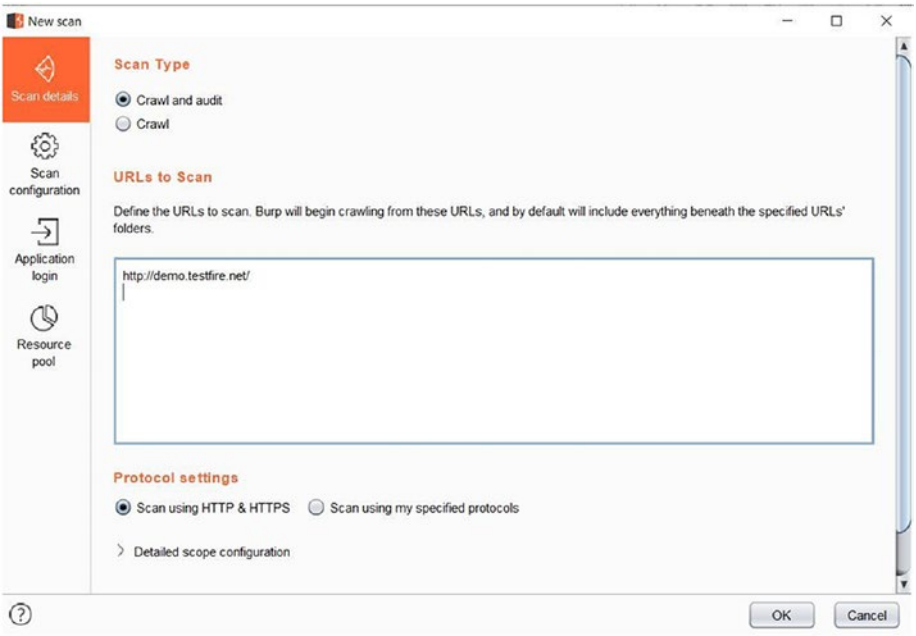

Рис.8-2. *Новая конфигурация сканирования*

Далее идет раздел области видимости, как показано на рисунке 8-3. URL-адреса, которые мы хотим включить в аудит, должны быть указаны на вкладке "Included URL prefixes", и если есть какие-либо конкретные URL-адреса, которые мы не хотим включать в аудит, их необходимо добавить в раздел "Excluded URL prefixes".

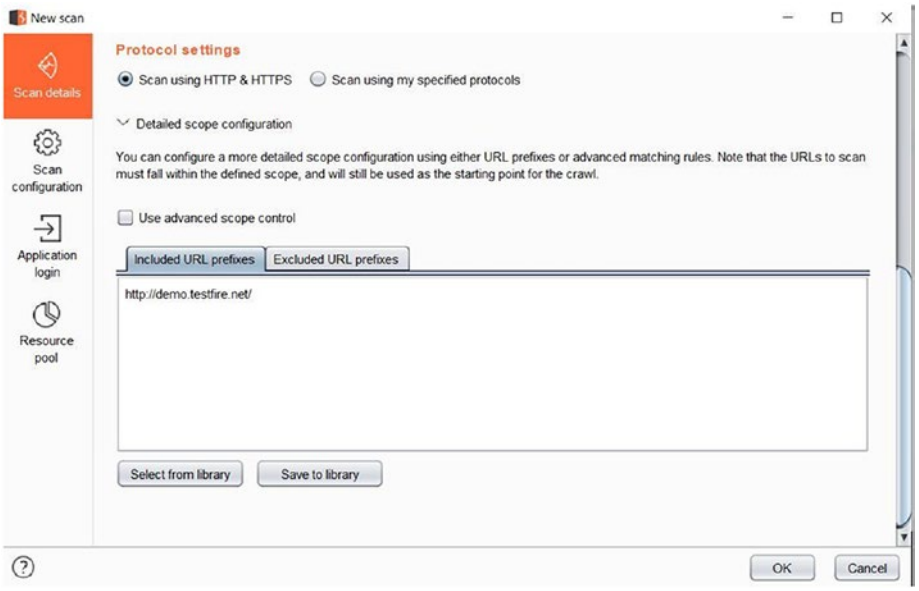

Рис.8-3. *Новая конфигурация сканирования*

Теперь, когда мы настроили целевой URL-адрес, нам просто нужно нажать "OK", и начнется обход и аудит, как показано на рисунке 8-4. Однако важно отметить, что это действие начнется с конфигурации сканирования по умолчанию.

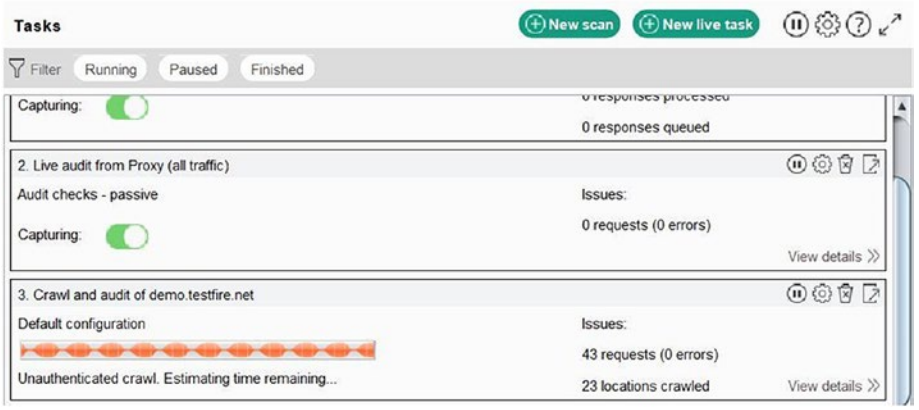

Рис.8-4. *Выполняемые задачи сканирования*

В следующем разделе мы рассмотрим настройку конфигурации сканирования.

Конфигурация сканирования

В предыдущем разделе мы настроили и инициировали задачу обхода и аудита по целевому URL-адресу, но с настройками конфигурации по умолчанию. В этом разделе мы рассмотрим, как настроить конфигурацию сканирования в соответствии с нашими потребностями. Чтобы настроить конфигурацию сканирования, перейдите на вкладку "Scan configuration", как показано на рис.8-5.

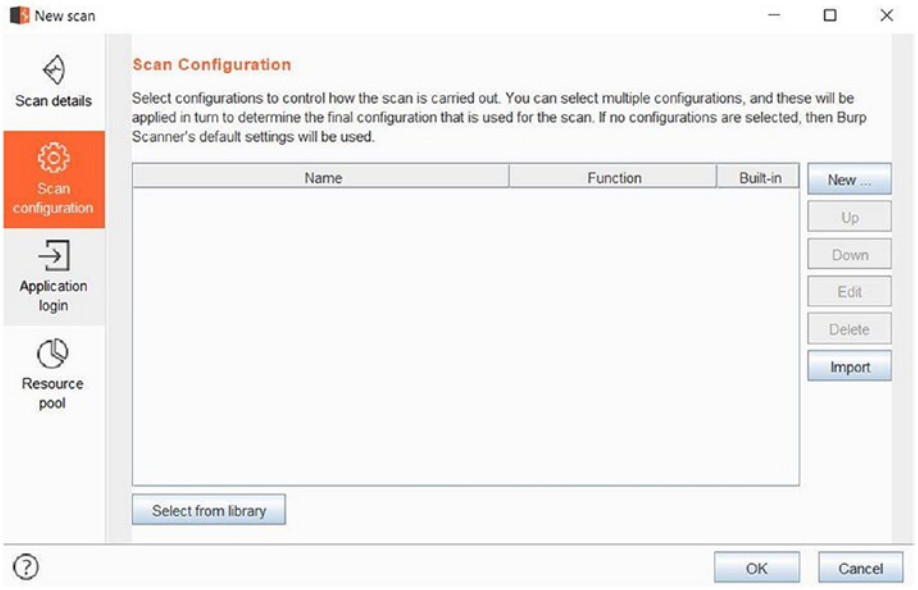

Рис.8-5. *Конфигурация сканирования*

Конфигурация сканирования позволяет нам настраивать параметры обхода, а также параметры аудита. Сначала мы рассмотрим параметры конфигурации обхода. Нажмите на кнопку "New' и выберите "Crawl". Появится новое окно, как показано на рисунке 8-6.

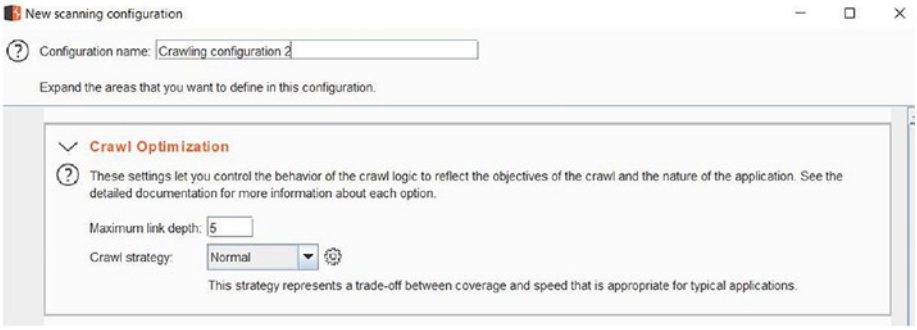

Рис.8-6. *Параметры оптимизации обхода*

Конфигурация оптимизации обхода позволяет нам установить максимальную глубину ссылки, до которой мы хотим выполнить обход, а также стратегию обхода, которая по умолчанию установлена в обычное состояние. Мы можем изменить стратегию обхода на быструю, выбрав ее в раскрывающемся меню, в зависимости от конкретного сценария сканирования.

Затем мы можем настроить ограничения по времени обхода, как показано на рисунке 8-7. Если целевое приложение большое и сложное, то для обхода может потребоваться много времени. Мы можем установить ограничение на это, определив максимальное время, которое мы хотим потратить на обход приложения. Мы также можем ограничить обход по количеству обнаруженных местоположений или максимальному количеству запросов, сделанных во время функции обхода.

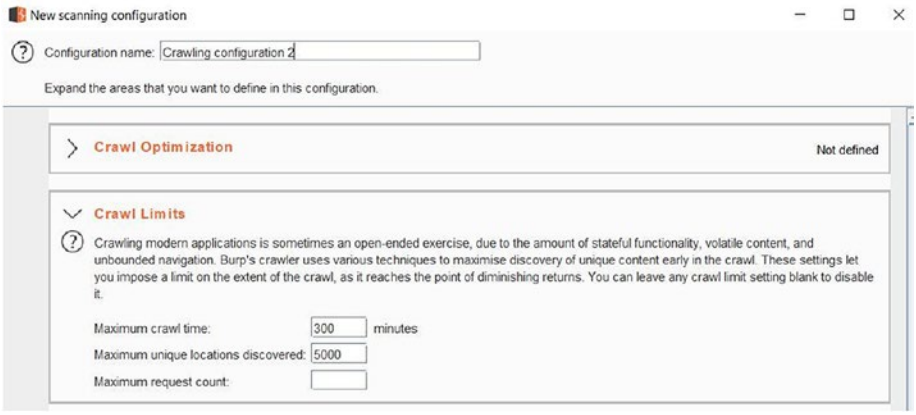

Рис.8-7. *Конфигурация ограничения обхода*

Следующий параметр конфигурации относится к функциям входа в систему, как показано на рисунке 8-8. Может случиться так, что целевое приложение имеет функцию входа в систему. В таком случае Burp Suite даже попытается зарегистрировать нового тестового пользователя.

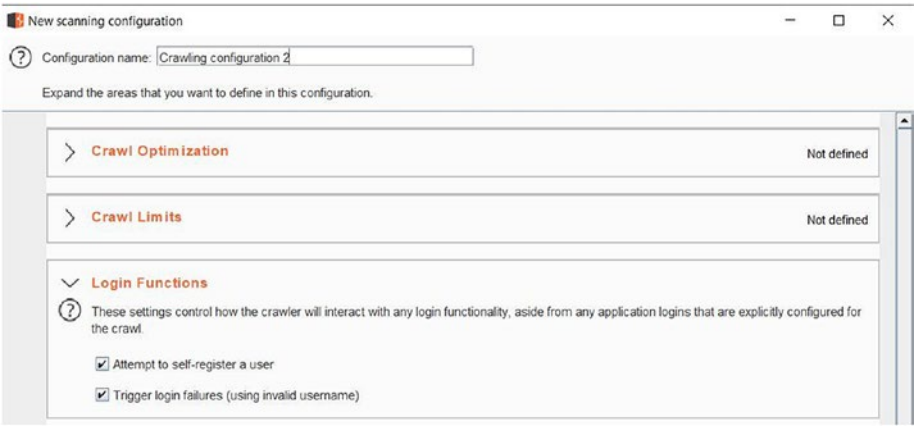

Рис.8-8. *Конфигурация функции входа в систему*

Следующий параметр конфигурации связан с обработкой ошибок приложения во время функции обхода, как показано на рисунке 8-9. Ошибки приложений могут быть вызваны несколькими причинами, такими как сбой аутентификации, проблемы с сетью и т.д. Этот параметр конфигурации указывает сканеру Burp Suite приостанавливать функцию обхода и аудита, если имеется определенное количество последовательных ошибок приложения.

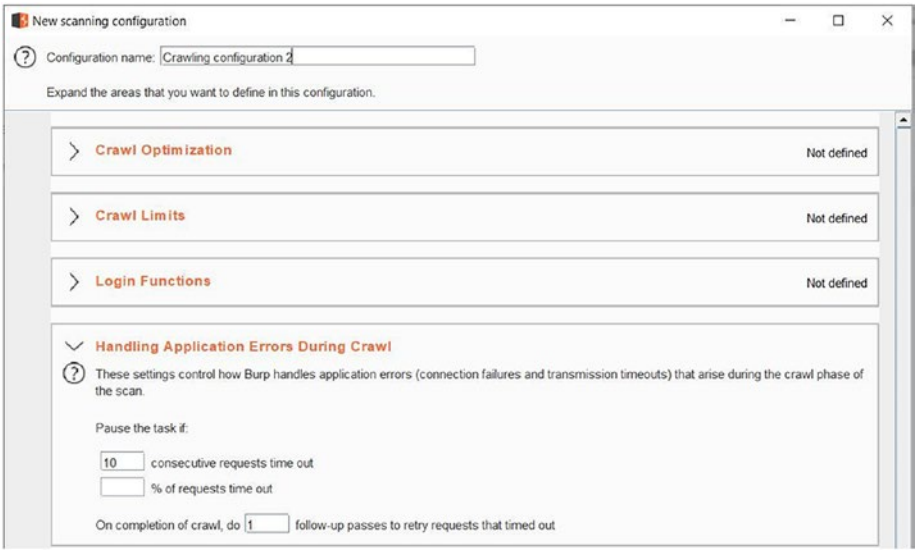

Рис.8-9. *Настройка ошибок приложений во время обхода*

Следующий набор параметров конфигурации отличается, как показано на рис. 8-10. Это включает в себя настройки того, хотим ли мы, чтобы сканер Burp Suite автоматически отправлял формы, или хотим ли мы настроить user-agent или хотим ли получить robots.txt, sitemap и т. д.

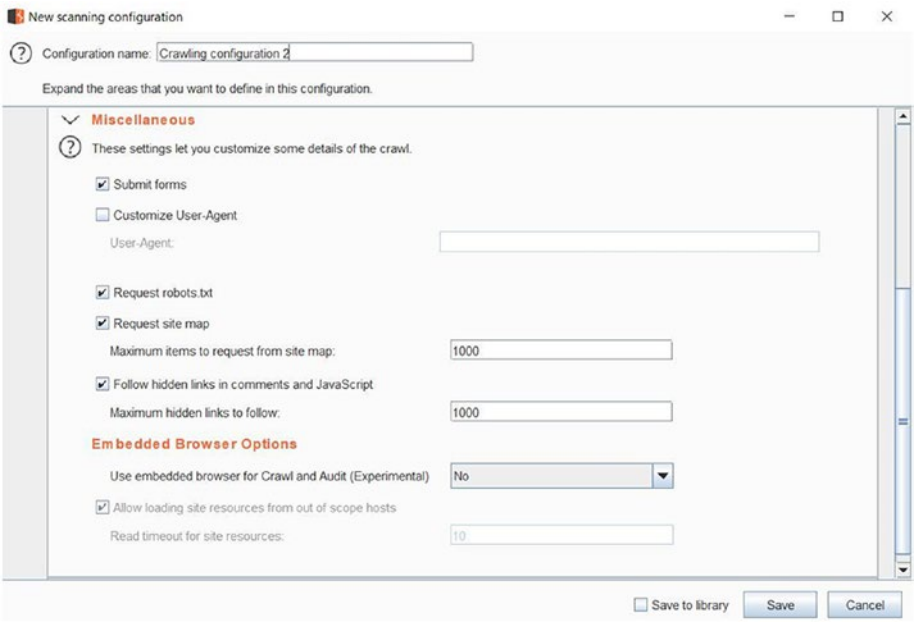

Рис.8-10. *Различные конфигурации обхода*

Следующий набор параметров конфигурации связан с функцией аудита. Начнем с того, что первым параметром конфигурации аудита является "Audit Optimization", как показано на рис. 8-11. Этот параметр позволяет нам настроить скорость и точность аудита. Скорость аудита может быть установлена на быструю, нормальную или тщательную, в то время как точность аудита может быть установлена на нормальную или для минимизации ложных срабатываний.

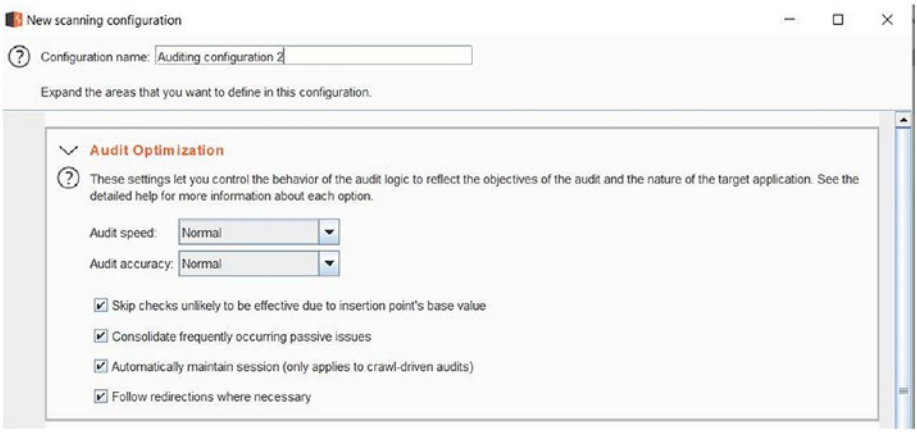

Рис.8-11. *Конфигурация оптимизации аудита*

Следующий параметр конфигурации аудита связан с типом проблем, о которых сообщается, как показано на рис.8-12. Сканер Burp Suite обнаруживает множество проблем. Однако во время конкретного сценария тестирования может случиться так, что потребуется протестировать только определенный тип проблемы. В таком случае не стоит тратить время на тестирование всех других типов проблем. Следовательно, этот параметр конфигурации позволяет нам настраивать тип проблем, которые мы хотим проверить во время сканирования.

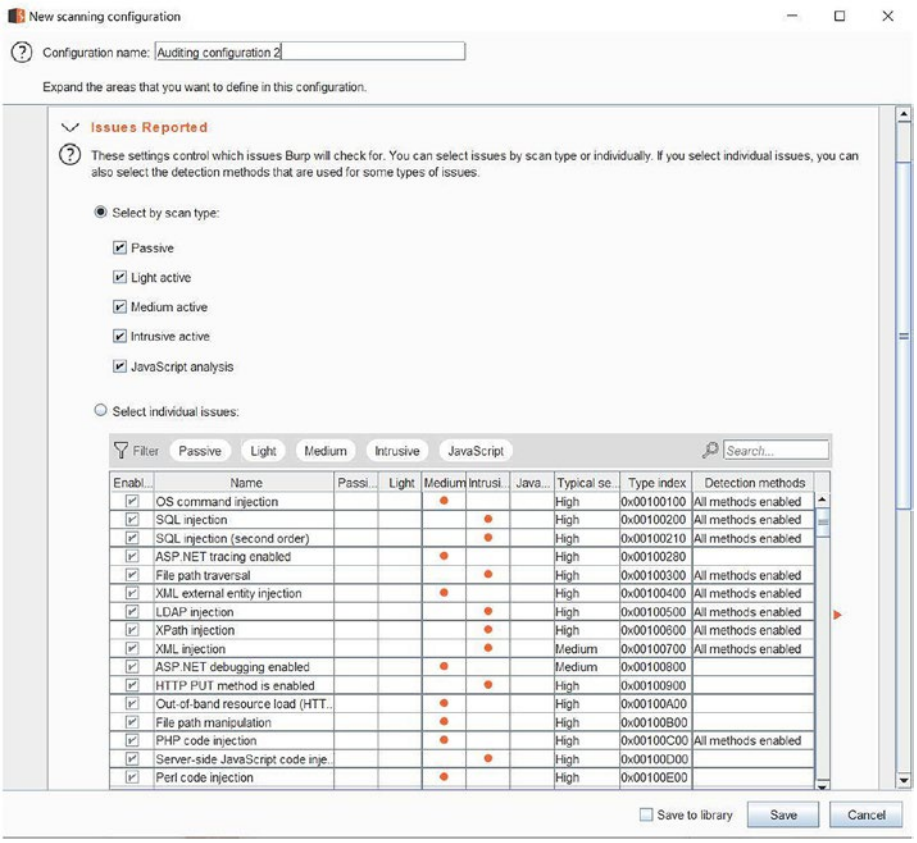

Рис.8-12. *Тип проблем, которые должны быть обнаружены во время аудита*

Следующий параметр конфигурации аудита связан с обработкой ошибок приложения во время выполнения функции аудита, как показано на рис.8-13. Мы уже видели аналогичный параметр конфигурации для функции обхода. Этот параметр помогает настроить количество сбоев, после которых задача аудита будет приостановлена.

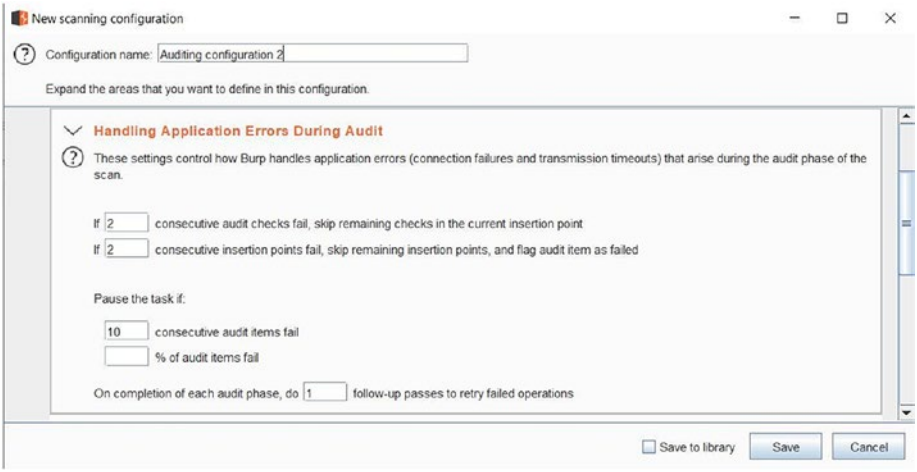

Рис.8-13. *Настройка ошибок приложений во время аудита*

Следующий параметр конфигурации аудита связан с типом точек вставки, которые мы хотим, чтобы сканер Burp Suite атаковал во время выполнения функции аудита, как показано на рис.8-14. Выбор всех типов точек вставки увеличит возможность обнаружения большего количества уязвимостей, но в то же время для завершения аудита потребуется больше времени.

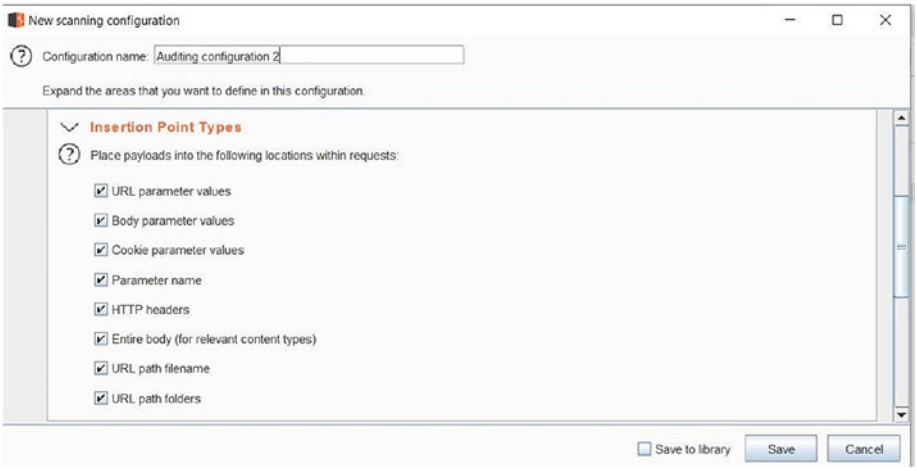

Рис.8-14. *Настройка типов точек вставки*

Все параметры конфигурации сканирования обхода и аудита, которые мы видели, по умолчанию имеют оптимальные значения. Мы можем быстро запустить новую задачу обхода и аудита, используя конфигурацию сканирования по умолчанию. Однако в зависимости от конкретных сценариев сканирования вам может потребоваться настроить и изменить параметры.

Вход в приложение

Следующим важным параметром конфигурации сканирования является настройка "Application login", как показано на рис.8-15. При сканировании целевого приложения мы можем наткнуться на определенные страницы, которые не требуют аутентификации, в то время как может быть несколько страниц, доступ к которым возможен только после аутентификации. Если мы хотим, чтобы сканер Burp Suite также проверял страницы, на которых выполняется аутентификация, нам необходимо предоставить учетные данные.

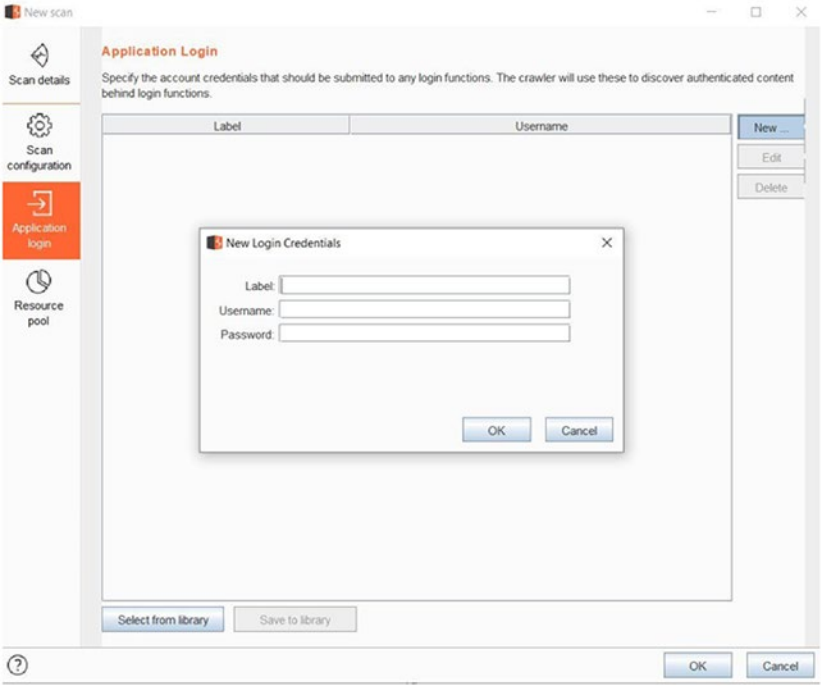

Рис.8-15. *Конфигурация входа в приложение*

Учетные данные можно добавить, просто нажав на кнопку "New" и указав необходимые имя пользователя и пароль.

Пулы ресурсов

Последний параметр конфигурации сканирования - “Resource Pool”, как показано на рис. 8-16. Пул ресурсов помогает определить системные ресурсы, которые будут использоваться для выполнения нескольких задач. По умолчанию создается пул ресурсов, который позволяет выполнять не более 10 одновременных запросов. Мы можем оставить это значение по умолчанию, если только мы не хотим выполнять многозадачность в рамках одного и того же проекта Burp Suite.

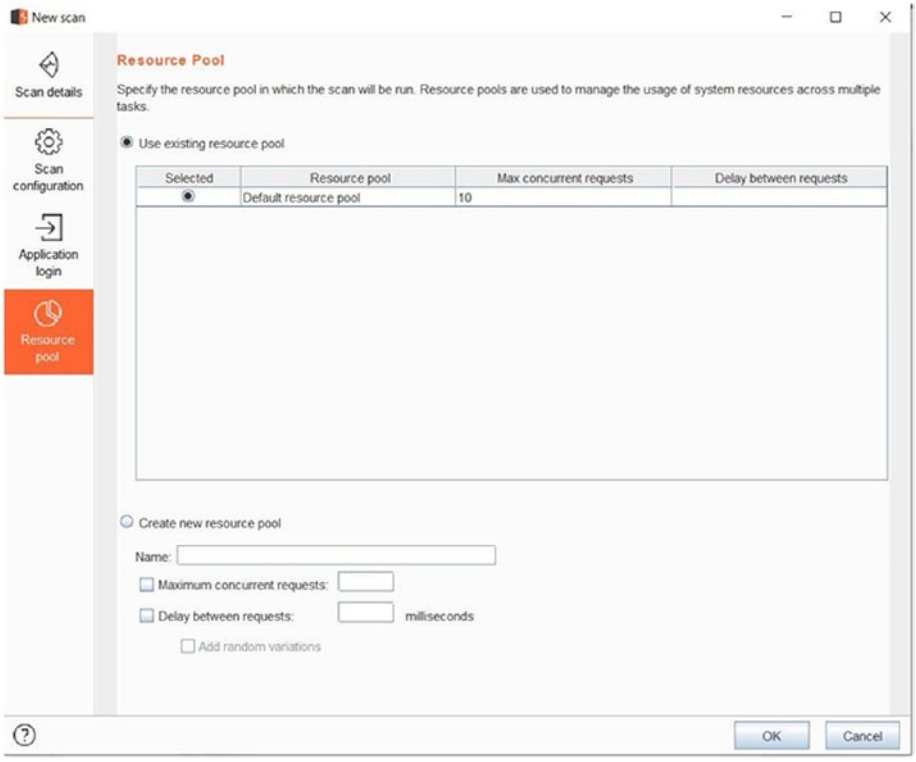

Рис.8-16. *Конфигурация пулов ресурсов*

Отчет

Сообщать о проблемах в презентабельном формате так же важно, как и находить их. Burp Suite предлагает отличную функцию отчетности, которая помогает нам создавать отчет в требуемом формате со всеми соответствующими сведениями об уязвимости. После создания, отчет может быть передан соответствующим заинтересованным сторонам для принятия дальнейших мер.

Как только задача обхода и аудита будет завершена, все проблемы, обнаруженные во время сканирования, будут перечислены в области "Issue activity", как показано на рис. 8-17. Теперь нам просто нужно выбрать проблемы, которые мы хотим включить в отчет. Для этого просто щелкните правой кнопкой мыши на проблеме, о которой необходимо сообщить, и нажмите "Report issue".

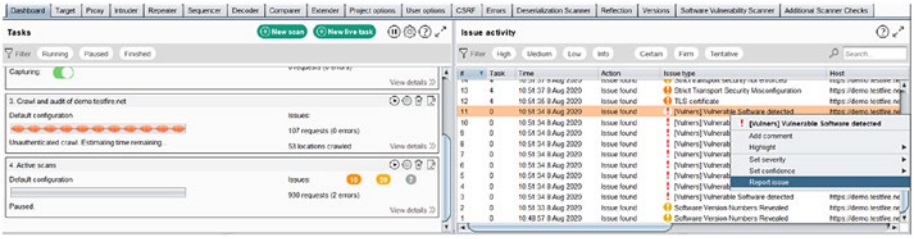

Рис.8-17. *Экспорт проблем для отчета*

Мастер отчетов Burp Suite теперь спросит нас о формате отчета, который мы хотим получить, как показано на рис.8-18. В настоящее время Burp Suite поддерживает создание отчетов в форматах HTML или XML.

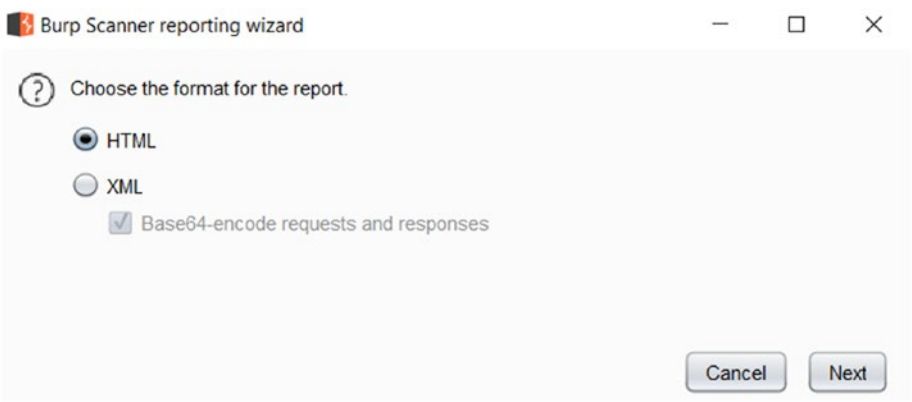

Рис.8-18. *Выбор формата для отчета*

Затем нам нужно выбрать, какие сведения о проблеме требуются в отчете, как показано на рис. 8-19.

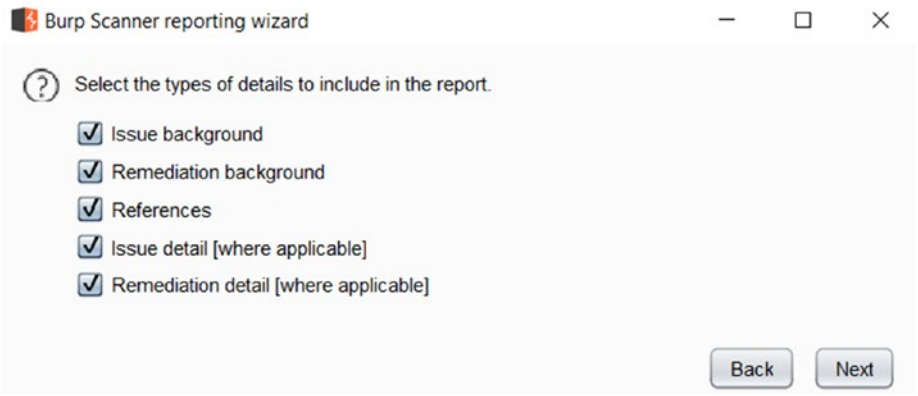

Рис.8-19. *Выбор типа сведений, которые будут включены в отчет*

Затем нам нужно выбрать, нужны ли нам полные HTTP-запросы и ответы на указанные проблемы или только соответствующие выдержки, как показано на рисунке 8-20.

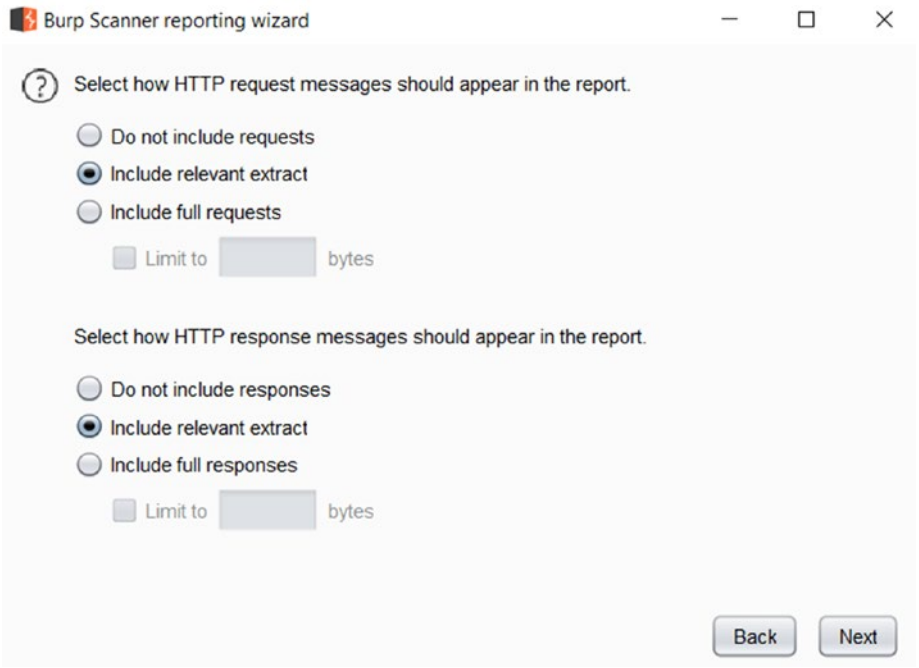

Рис.8-20. *Выбор формата запросов и ответов для отчета*

Наконец, нам нужно выбрать имя и местоположение, в котором мы хотим создать отчет, а также название отчета, как показано на рис. 8-21.

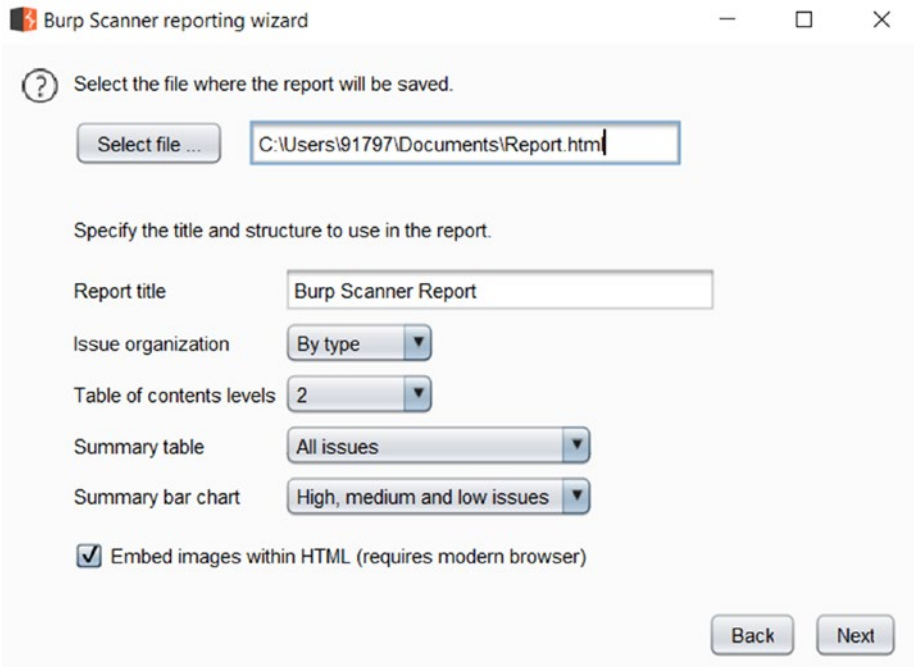

Рис.8-21. *Настройка местоположения, в котором будет сохранен отчет*

Теперь мастер отчетов Burp Suite создаст отчет об уязвимости, как показано на рис.8-22.

Рис.8-22. *Создание отчета*

Созданный отчет можно просмотреть в любом из браузеров, как показано на рис. 8-23. В отчете содержится краткое изложение выводов, основанных на достоверности, а также на серьезности.

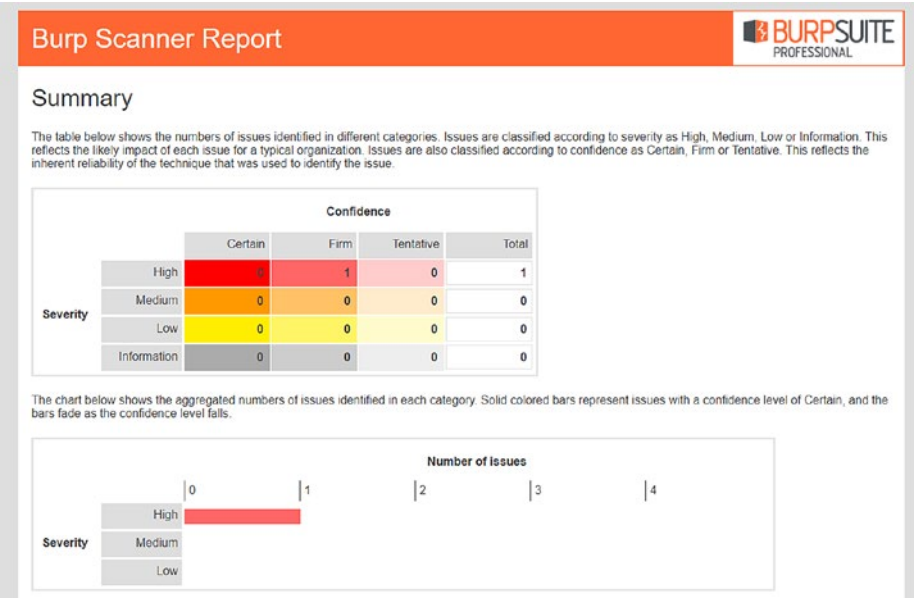

Рис.8-23. *Просмотр отчета в браузере*

В отчете также подробно показана уязвимость вместе с соответствующим запросом и ответом, как показано на рисунке 8-24.

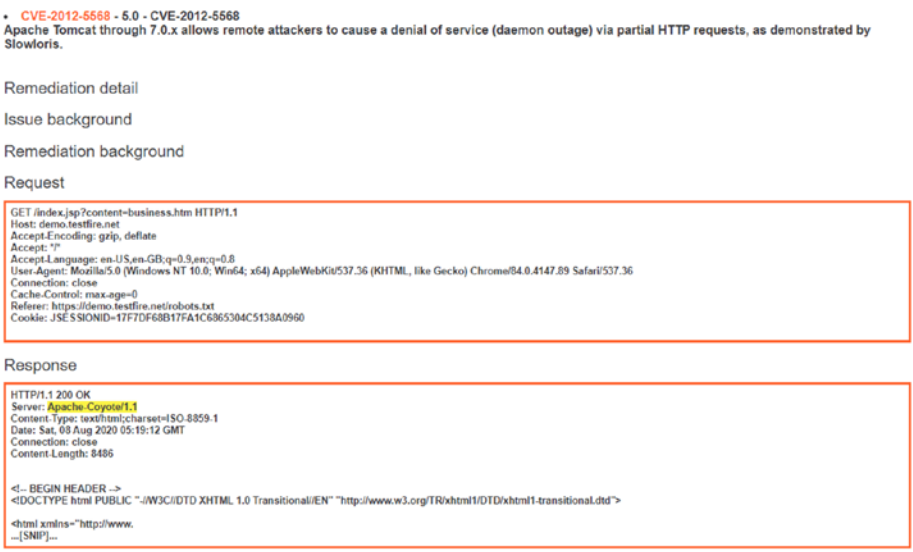

Рис.8-24. *Подробная информация об уязвимости в отчете*

Резюме

В этой главе мы узнали о сканере Burp Suite и о том, как его можно настроить для эффективного автоматического поиска уязвимостей приложений.

В следующей главе мы рассмотрим, как использовать Extender Burp Suite для установки дополнительных плагинов и расширения возможностей.

Упражнения

1. Отсканируйте любое из целевых веб-приложений, используя функцию обхода и аудита Burp Suite.
2. Создайте отчет в формате HTML о проблемах, обнаруженных во время сканирования.

Глава 9

Расширения Burp Suite

В предыдущей главе мы узнали о сканере Burp Suite, который эффективно помогает автоматизировать обнаружение уязвимостей. В этой главе мы рассмотрим функцию "Extender" в Burp Suite, с помощью которой мы сможем еще больше расширить возможности Burp Suite.

Расширения Burp Suite

На протяжении всей книги мы рассматривали различные возможности Burp Suite как для ручного, так и для автоматического обнаружения уязвимостей. Мы изучили различные инструменты и утилиты в Burp Suite, которые можно использовать для конкретных задач.

Burp Suite теперь больше похож на платформу, которая достаточно гибка для размещения внешних функций и утилит. Как мы уже видели, Burp Suite предоставляет множество возможностей "из коробки". Однако эти возможности могут быть расширены в дальнейшем с помощью расширений.

Расширения Burp Suite представлены в различных формах, как показано ниже:

Расширения по умолчанию – Эти расширения перечислены по умолчанию, из коробки в любом из наборов Burp Suite, и могут быть установлены с помощью Extender Burp Suite.

Pro расширения – Это расширения, которые могут быть установлены и запущены только в версии Burp Suite Professional Edition.

S. Rahalkar, *A Complete Guide to Burp Suite*,
https://doi.org/10.1007/978-1-4842-6402-7_9

Обычные расширения – Это расширения, которые могут быть установлены и запущены в Burp Suite Community Edition, а также в Professional Edition.

Другие расширения – Burp Suite открыл API, для которых разработчики могут создавать новые пользовательские расширения. Такие расширения не являются частью официального магазина расширений, но их необходимо загрузить и установить вручную.

Магазин BApp

Самый простой способ установить расширение в Burp Suite - через магазин BApp. Чтобы получить доступ к магазину BApp, просто перейдите в раздел Extender > BApp Store, как показано на рис.9-1.

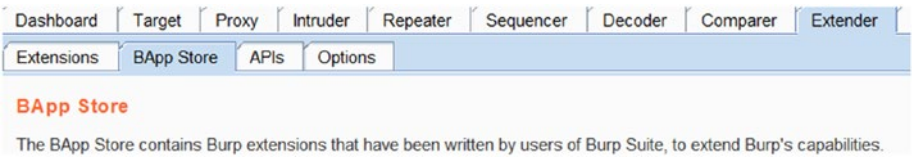

Рис.9-1. *Магазин BApp*

Магазин BApp имеет очень простой в использовании интерфейс с двумя панелями, как показано на рис. 9-2.

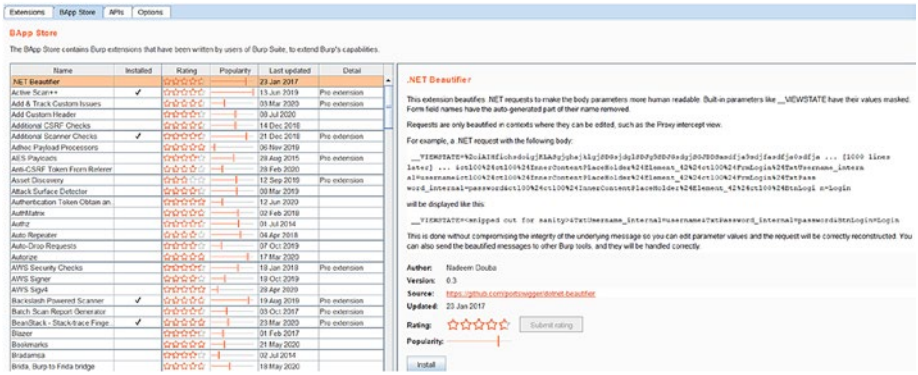

Рис.9-2. *Магазин BApp*

На левой панели перечислены все доступные расширения, а также следующая информация:

- Название расширения,
- Установлен ли он в данный момент или нет,
- Рейтинг расширения,
- Популярность расширения,
- Дата последнего обновления расширения,
- Доступно расширение только для использования в Burp Suite Professional Edition или его можно также использовать в Community Edition.

На правой панели отображается подробная информация о любом из расширений, которые мы выбираем на левой панели. Это включает в себя следующую информацию:

- Подробная информация о том, что это за расширение и как его можно использовать
- Автор
- Версия расширения

- Источник
- Дата последнего обновления расширения
- Рейтинг и популярность расширения
- Кнопка "Install" для установки и добавления расширения в текущую настройку Burp Suite

Важно отметить, что новые расширения продолжают регулярно добавляться в магазин BApp. Чтобы убедиться, что список расширений является последним, просто нажмите кнопку "Refresh list", как показано на рис. 9-3.

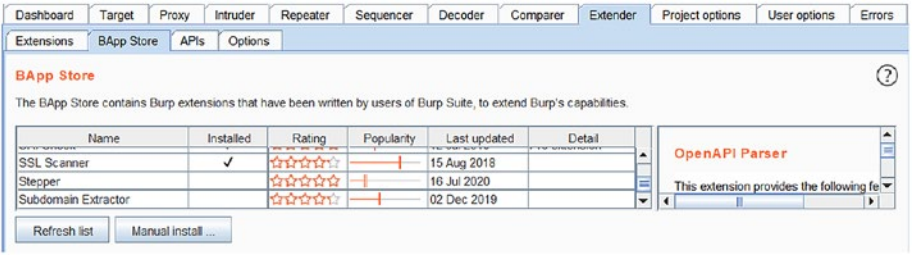

Рис.9-3. *Просмотр расширений в магазине BApp*

Ниже приведены некоторые полезные расширения из магазина BApp:

- **Active Scan++** – Это расширение разработано для дополнительного расширения возможностей пассивного и активного сканирования Burp Suite.
- **Additional Scanner Checks** – Это расширение добавляет еще несколько проверок к пассивному сканеру, например, DOM-based XSS и т. д.
- **CSRF Scanner** – Это расширение помогает пассивно сканировать уязвимости для подделки межсайтовых запросов (CSRF).

- **Discover Reverse Tabnabbing** – Это расширение ищет в HTML-коде возможные уязвимости Tabnabbing.
- **Error Message Checks** – Это расширение помогает пассивно обнаруживать любые сообщения об ошибках или исключениях, которые могут содержать конфиденциальную информацию, например, трассировку стека.
- **Headers Analyzer** – Это расширение пассивно проверяет заголовки ответа и отмечает все отсутствующие заголовки безопасности, такие как X-XSS-Protection, X-Frame-Options и многие другие.
- **HTML5 Auditor** – Это расширение проверяет, не были ли использованы какие-либо потенциально опасные функции HTML5, такие как хранение конфиденциальных данных в хранилище на стороне клиента, геолокация клиента и т.д.
- **J2EEScan** – Это расширение помогает улучшить покрытие тестов для J2EE-приложений, а также добавляет дополнительные тестовые сценарии.
- **Java Deserialization Scanner** – Это расширение расширяет возможности Burp Suite по обнаружению уязвимостей десериализации Java.
- **JavaScript Security** – Это расширение добавляет несколько пассивных проверок, связанных с безопасностью JavaScript, таких как проблемы DOM, Cross-Origin Resource Sharing (CORS) и т.д.
- **Retire.js** – Это расширение пассивно отслеживает трафик и обнаруживает использование любой уязвимой сторонней библиотеки вместе с необходимой информацией о CVE.
- **SameSite Reporter** – Это расширение проверяет, был ли установлен атрибут SameSite в cookies или нет.

- **Software Version Reporter** – Это расширение пассивно анализирует трафик и сообщает все данные о версии программного обеспечения. Эта информация может помочь в перечислении приложений.
- **Upload Scanner** – Это расширение добавляет возможности в Burp Suite для обнаружения функций загрузки файлов и связанных с ними уязвимостей.
- **Web Cache Deception Scanner** – Это расширение сканирует приложение на наличие уязвимости Web Cache Deception.
- **CSP Auditor** – Это расширение сканирует заголовки ответа и проверяет, правильно ли настроена политика безопасности содержимого (CSP) или нет.
- **CVSS Calculator** – Это расширение позволяет оценивать уязвимости по методологии CVSS в Burp Suite.

Ручная установка

В предыдущем разделе мы рассмотрели, как можно просматривать, выбирать и устанавливать расширения с помощью магазина BApp. Не все написанные расширения доступны в магазине BApp. Могут существовать расширения, написанные отдельными авторами, опубликованные на различных сайтах, таких как GitHub и т.д. В таком случае, если расширение не присутствует в магазине BApp, нам нужно скачать его отдельно и установить вручную. Чтобы установить расширения вручную, перейдите на вкладку 'Extensions' в Extender, как показано на рисунке 9-4.

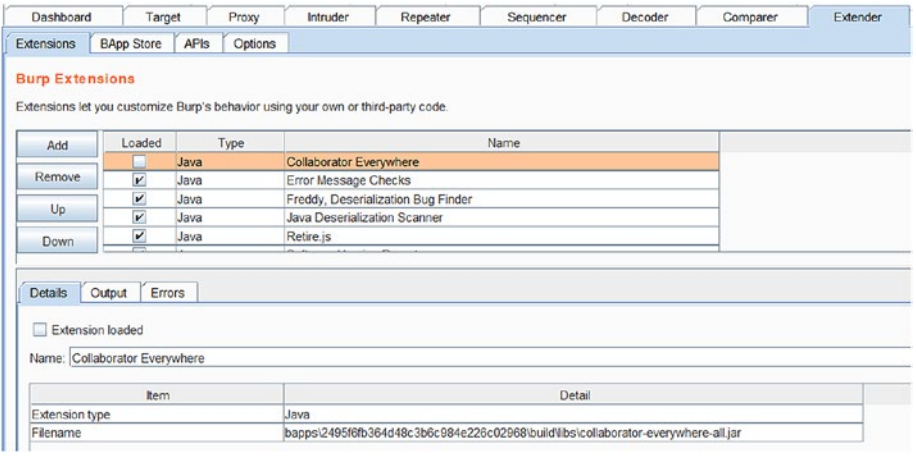

Рис.9-4. *Добавление расширений вручную*

Burp Suite принимает установку сторонних расширений со следующими форматами, как показано на рисунке 9-5.

- Java
- Python
- Ruby

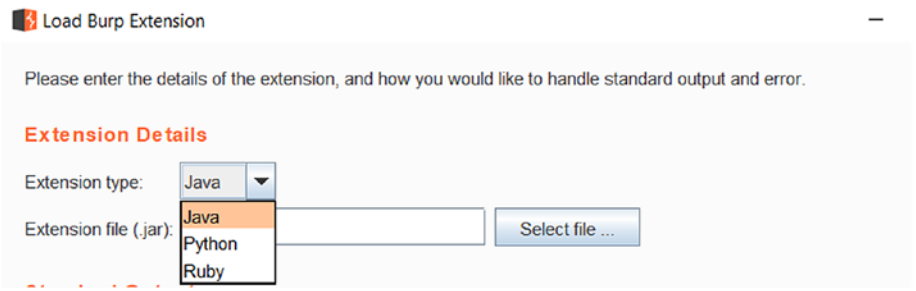

Рис.9-5. *Выбор типа расширения*

Другие опции включают, хотим ли мы показать вывод и ошибки после установки расширения на консоли или сохранить их в файл, как показано на рисунке 9-6.

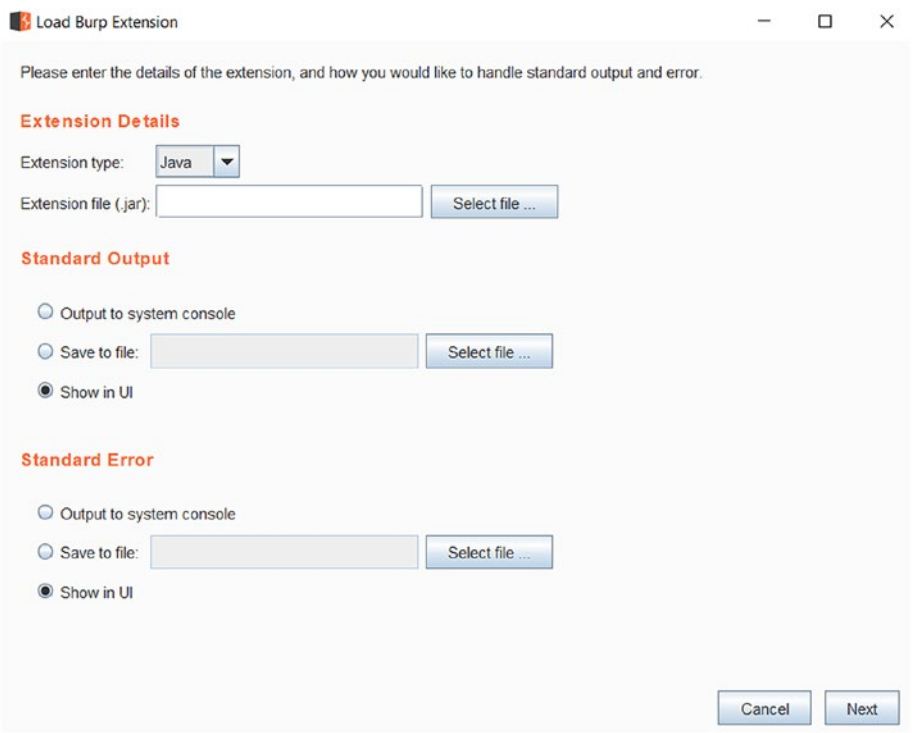

Рис.9-6. *Добавление расширений вручную*

Чтобы установить расширение, выберите тип расширения (Java / Python / Ruby), а затем просто найдите и выберите место, где расширение находится на диске, как показано на рисунке 9-7.

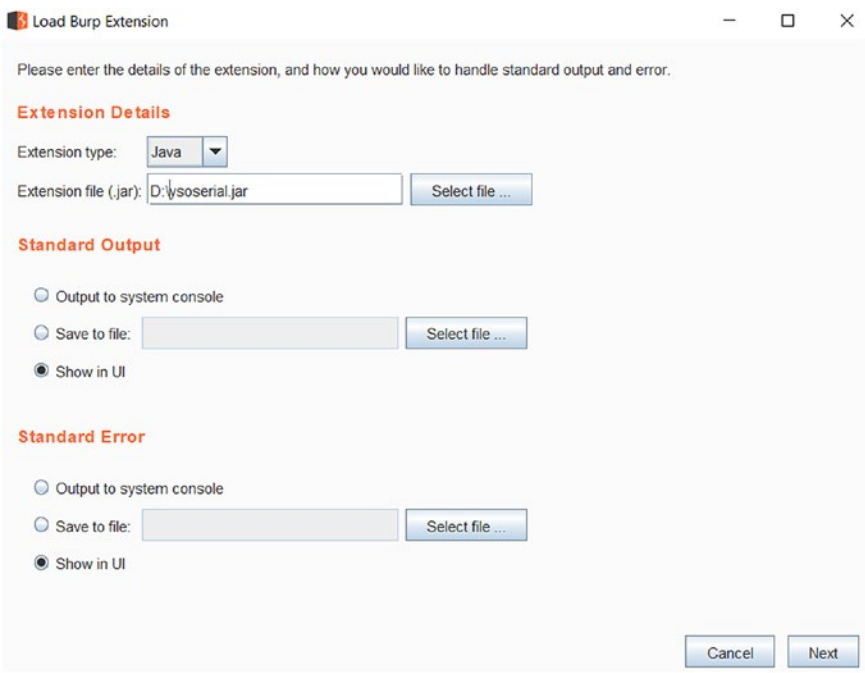

Рис.9-7. *Выбор файла расширения*

Если установка расширения завершена успешно, появится сообщение, как показано на рисунке 9-8.

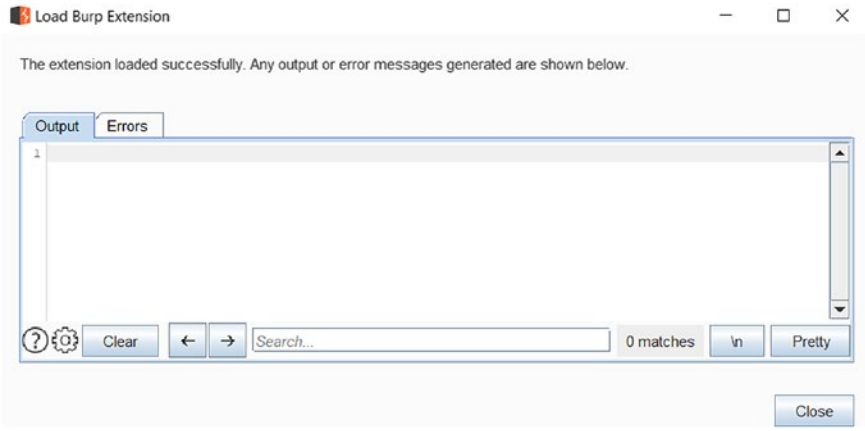

Рис.9-8. *Загрузка расширений вручную*

Настройки

Теперь, когда мы увидели, как устанавливать расширения либо через магазин BApp, либо вручную, давайте рассмотрим некоторые дополнительные настройки, связанные с расширениями.

- Доступ к настройкам можно получить, перейдя на вкладку 'Extender > Options', как показано на рисунке 9-9. Первые две настройки определяют, хотите ли вы автоматически перезагружать расширения при запуске Burp Suite и хотите ли вы автоматически обновлять расширения при запуске.

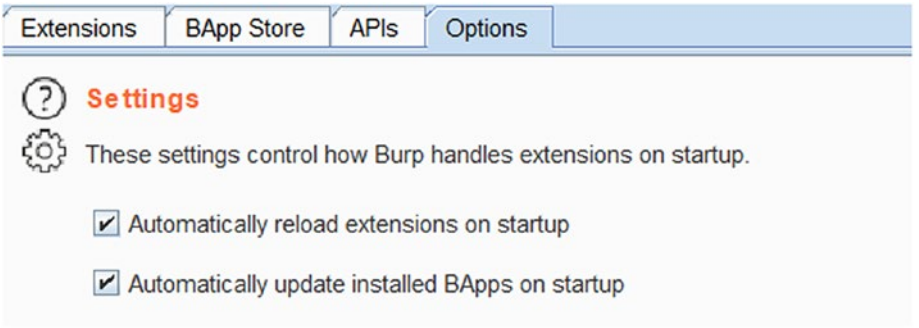

Рис.9-9. *Опции магазина BApp*

Следующая настройка связана со средой Java. Большинство расширений написано на языке Java. Для обеспечения правильной работы этих расширений может потребоваться указать путь к любым дополнительным библиотечным зависимостям, как показано на рисунке 9-10.

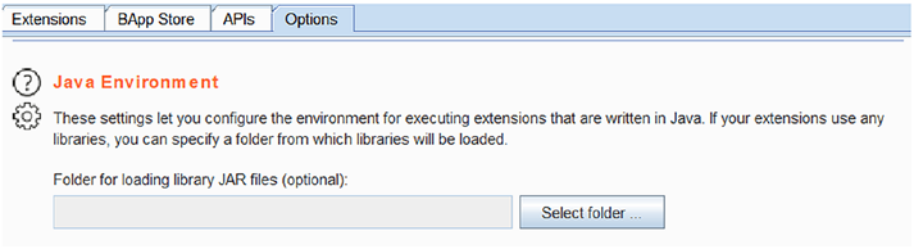

Рис.9-10. *Настройка среды Java*

Следующая установка связана с настройкой среды Python. Для некоторых расширений требуется интерпретатор Python, реализованный на Java под названием Jython. Jython можно загрузить и установить с сайта https://www.jython. org/download. После установки необходимо обновить путь к директории установки Jython, как показано на рисунке 9-11.

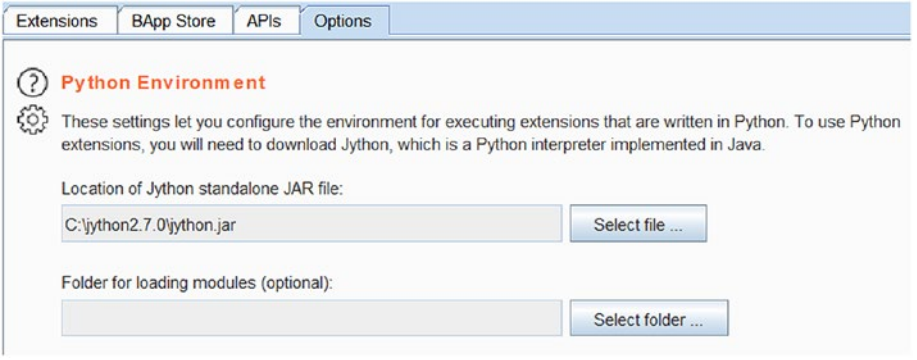

Рис.9-11. *Настройка среды Python*

Последний параметр связан с настройкой среды Ruby. Как мы видели ранее, Burp Suite поддерживает расширения, написанные на Ruby, поэтому нам нужно указать путь к интерпретатору Ruby, как показано на рисунке 9-12. Чтобы запустить расширение в Burp Suite, написанное на Ruby, необходимо загрузить и установить JRuby с сайта https://www.jruby.org/download.

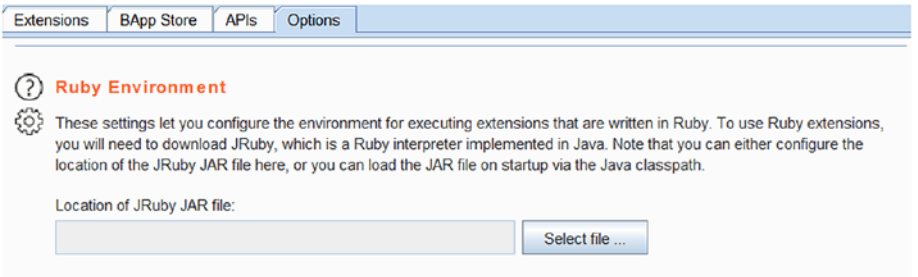

Рис.9-12. *Настройка среды Ruby*

Другие полезные расширения

Ранее в этой главе мы уже познакомились с некоторыми полезными расширениями, доступными в магазине BApp. Поскольку Burp Suite предоставляет разработчикам интерфейс прикладного программирования (API), можно легко писать и разрабатывать пользовательские расширения по мере необходимости. Вот несколько дополнительных полезных расширений, которые можно вручную установить в Burp Suite.

- **sometime** – Это расширение можно загрузить с сайта https://github.com/linkedin/sometime. Это расширение пассивно отслеживает трафик, чтобы проверить, уязвимо ли приложение к Same Origin Method Execution.
- **burp-suite-gwt-scan** – Это расширение можно загрузить с сайта https://github.com/augustd/burp-suite-gwt-scan - Это расширение помогает автоматически определять точки вставки для GWT (Google Web Toolkit) запросов при отправке их активному сканеру или Burp Intruder.
- **Admin panel finder** – Это расширение можно загрузить с сайта https://github.com/moeinfatehi/Admin-Panel_Finder - Это расширение помогает в перечислении интерфейсов администратора инфраструктуры и приложений, которые могли быть оставлены открытыми по ошибке.

- **Pwnback –** Это расширение можно загрузить с сайта https://github.com/P3GLEG/PwnBack. Это расширение помогает получить старые и архивные версии приложения, если таковые имеются. Это может быть полезно для сравнения старой и текущей версий приложения, чтобы проверить изменения и связанные с ними уязвимости.
- **Minesweeper –** Это расширение можно загрузить с сайта https://github.com/codingo/Minesweeper - Это расширение помогает обнаружить скрипты, загружаемые с более чем 23000+ вредоносных доменов для добычи криптовалюты (криптоджекинг).

Дополнительный и полный список расширений Burp Suite приведен на сайте https://github.com/snoopysecurity/awesome-burp-extensions.

API

На протяжении всей этой главы мы видели использование Extender для добавления и установки новых расширений, которые значительно улучшают возможности Burp Suite. Burp Suite предлагает еще одну полезную функцию в виде интерфейса прикладного программирования (API). Используя эти API, можно писать собственные расширения. Список доступных API и подробное руководство по их использованию доступно на вкладке 'Extender > APIs', как показано на рисунке 9-13.

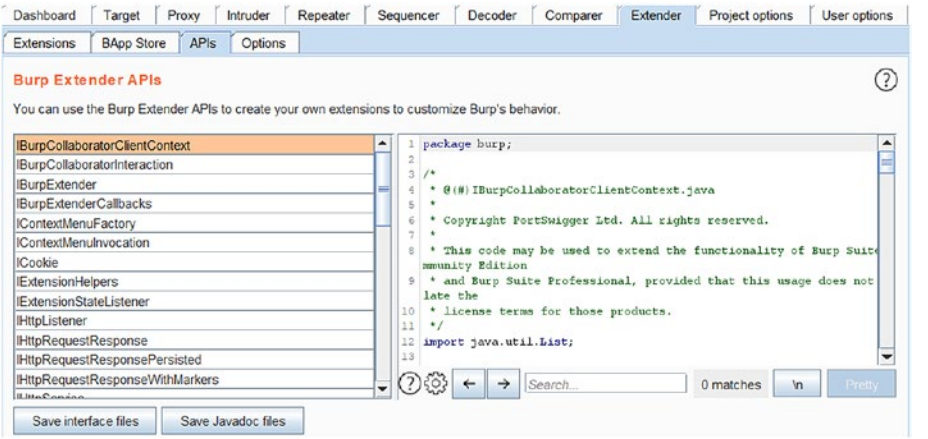

Рис.9-13. *API в Extender Burp Suite*

Резюме

В этой главе мы познакомились с Extender Burp Suite, который позволяет расширить возможности Burp Suite с помощью внешних расширений. Мы изучили магазин BApp, в котором есть список многих полезных расширений, а также научились устанавливать расширение вручную, если его нет в магазине BApp. Наконец, мы перечислили несколько дополнительных расширений, помимо тех, которые официально представлены в магазине BApp.

В следующей главе мы рассмотрим, как можно использовать возможности Burp Suite для тестирования мобильных приложений и API. Мы также рассмотрим полный рабочий процесс тестирования приложения с помощью Burp Suite.

Упражнения

1. Используйте магазин BApp для установки расширений, обсуждаемых в этой главе.
2. Выполните активное сканирование целевого приложения до установки расширений и после установки расширений. Обратите внимание на разницу в уязвимостях, найденных при обоих сканированиях.
3. Изучите дополнительные расширения, рассмотренные в этой главе, и попробуйте установить их вручную.

Глава 10

Тестирование мобильных приложений и API с помощью Burp Suite

В прошлой главе мы узнали о функции Extender в Burp Suite, которая позволяет расширить возможности Burp Suite с помощью сторонних расширений. В этой последней главе мы рассмотрим, как Burp Suite можно использовать для тестирования API и мобильных приложений. В заключение мы кратко опишем рабочий процесс тестирования любого веб-приложения с помощью Burp Suite.

Тестирование безопасности API с помощью Burp Suite

На протяжении всей этой книги мы знакомились с различными возможностями Burp Suite, которые можно использовать для тестирования безопасности веб-приложений. Однако сегодня современные приложения являются более функциональными и взаимосвязанными. Это достигается за счет использования интерфейсов прикладного программирования (API).

S. Rahalkar, *A Complete Guide to Burp Suite*,
https://doi.org/10.1007/978-1-4842-6402-7_10

Использование API значительно помогает автоматизировать задачи, но в то же время создает риски безопасности, если не реализовать его безопасно. Большинство уязвимостей веб-приложений применимы к API, но есть несколько уязвимостей, которые характерны именно для API. OWASP опубликовал список 10 лучших уязвимостей для API, который можно найти на сайте https://owasp.org/www-project-api-security/.

Подход к тестированию безопасности API с помощью Burp Suite очень похож на тестирование обычных веб-приложений, которые мы рассматривали в этой книге. Поскольку API взаимодействуют по протоколу HTTP/HTTPS, трафик может быть перехвачен и изменен в Burp Suite так же, как и любой другой запрос и ответ обычного веб-приложения. Для проведения тестирования безопасности API с помощью Burp Suite мы можем использовать один из следующих подходов:

1. Регулярно обходите по приложению и выявляйте конечные точки, принадлежащие API. После определения конечных точек API соответствующие запросы могут быть отправлены в Repeater или в Intruder для дальнейшего тестирования.

2. Во многих случаях, API вызываются через функциональные возможности пользовательского интерфейса (UI) в приложении. В таком случае можно просто создать новую задачу 'Crawl and Audit' в сканере Burp Suite и убедиться, что все проверки и задачи сканера выполнены.

3. Может существовать набор API, которые не вызываются напрямую ни из одного пользовательского интерфейса. Такие API часто тестируются вручную с помощью таких инструментов, как Postman. Мы можем легко интегрировать Postman с Burp Suite для сбора всего необходимого трафика API. Как только необходимые API-запросы и ответы попадают в Burp Suite, остается только протестировать их дальше с помощью Repeater или Intruder.

Сейчас мы рассмотрим, как можно интегрировать Postman с Burp Suite. Postman - это популярный инструмент, используемый для ручного тестирования API. Его можно скачать с сайта https://www.postman.com/downloads/. Интерфейс приложения Postman показан на рисунке 10-1.

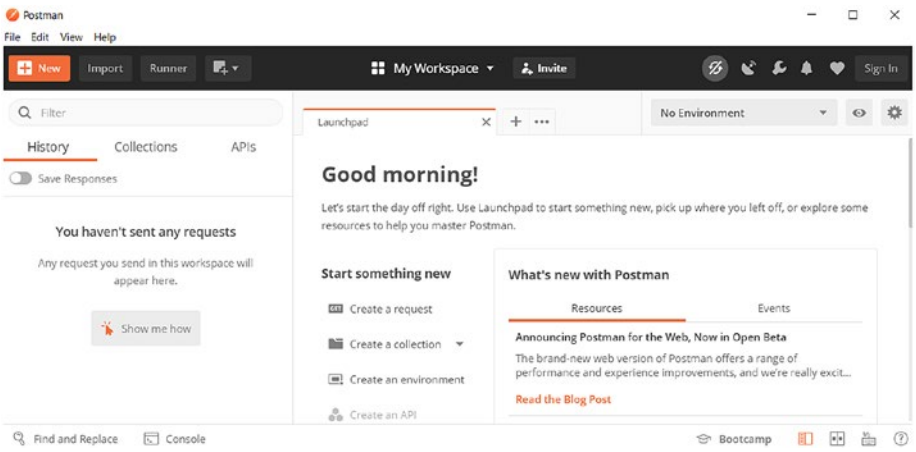

Рис.10-1. *Инструмент Postman*

Для того чтобы настроить Postman для работы вместе с Burp Suite, нажмите на опцию 'Settings' в правом верхнем углу, как показано на рисунке 10-2.

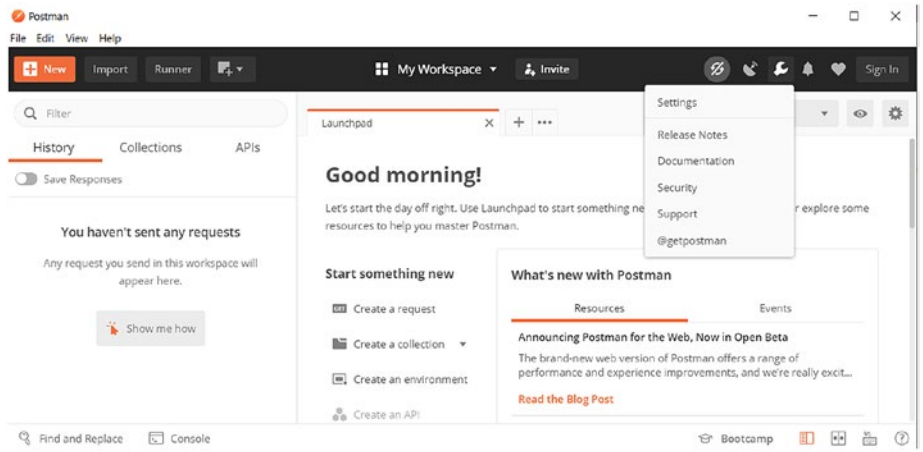

Рис.10-2. *Переход к настройкам в инструменте Postman*

Откроется новое окно настроек, как показано на рисунке 10-3.

Теперь перейдите на вкладку 'Proxy' и выберите 'Add a custom proxy configuration' и введите адрес прокси сервера как адрес машины, на которой запущен Burp Suite (обычно localhost или 127.0.0.1) и порт как 8080 или тот, на котором мы хотим, чтобы прокси слушатель Burp Suite был активен.

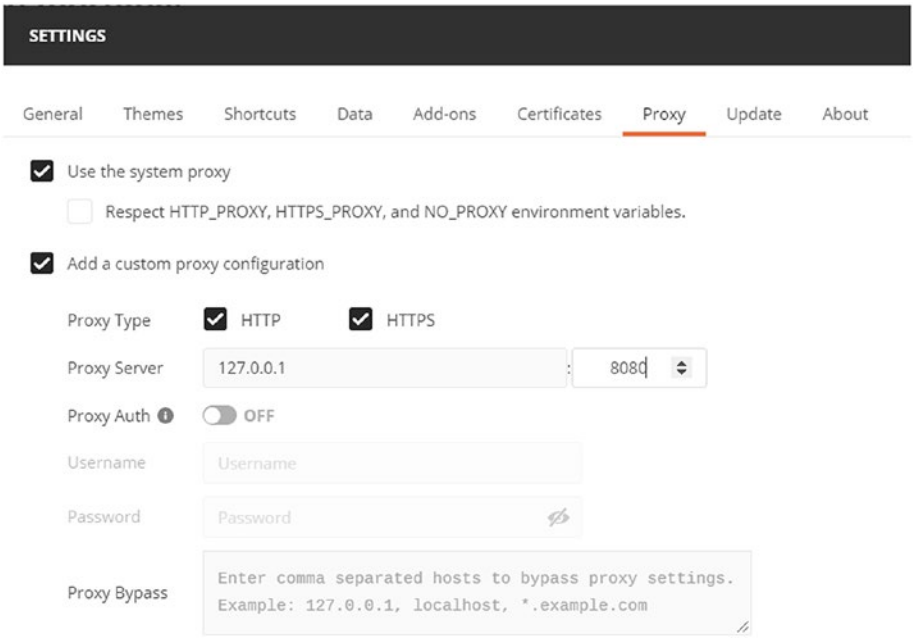

Рис.10-3. *Настройка прокси в инструменте Postman*

Теперь, когда мы закончили с настройкой на стороне Postman, нам нужно убедиться, что правильная конфигурация прокси также выполнена на стороне Burp Suite. Чтобы убедиться, что правильный прокси настроен в Burp Suite, перейдите на вкладку Proxy и Options, как показано на рисунке 10-4, и убедитесь, что IP-адрес и номер порта совпадают с тем, что было настроено ранее в Postman.

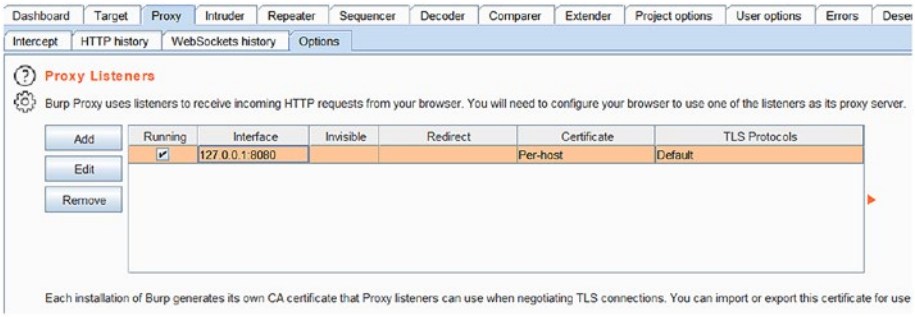

Рис.10-4. *Настройка прокси-слушателя Burp Suite*

После того как Postman и Burp Suite настроены на совместную работу, весь трафик, генерируемый Postman, будет проходить через Burp Suite. Это очень похоже на то, как мы настраивали наши браузеры для совместной работы с Burp Suite.

Теперь, когда мы увидели, как настроить Postman для работы вместе с Burp Suite, посмотрим, как работает обратный процесс; то есть, как экспортировать данные из Burp Suite в Postman для выборочного тестирования?

Хотя Burp Suite Repeater и Intruder служат для большинства целей тестирования безопасности API, может возникнуть необходимость протестировать конкретный API в интерфейсе Postman. В таком случае можно экспортировать запрос API из Burp Suite в инструмент Postman.

Для экспорта API-запроса из Burp Suite в Postman нам потребуется установить расширение под названием 'Postman Integration'. Просто перейдите на вкладку Extender, откройте 'BApp Store' и установите расширение 'Postman Integration', как показано на рисунке 10-5.

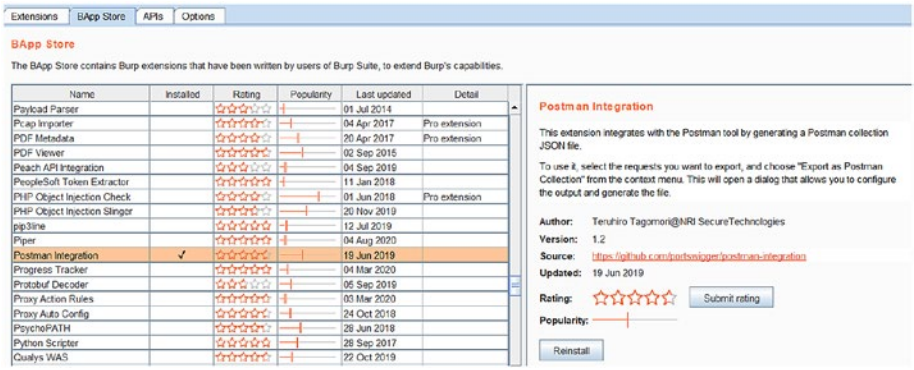

Рис.10-5. *Установка расширения Postman Integration в Burp Suite через BApp Store*

После установки расширения:

1. Перейдите на вкладку Target,
2. Выберите запрос, который вы хотите экспортировать в Postman,
3. Щелкните правой кнопкой мыши по запросу, который необходимо экспортировать,
4. Выберите опцию 'Export as Postman Collection', как показано на рисунке 10-6.

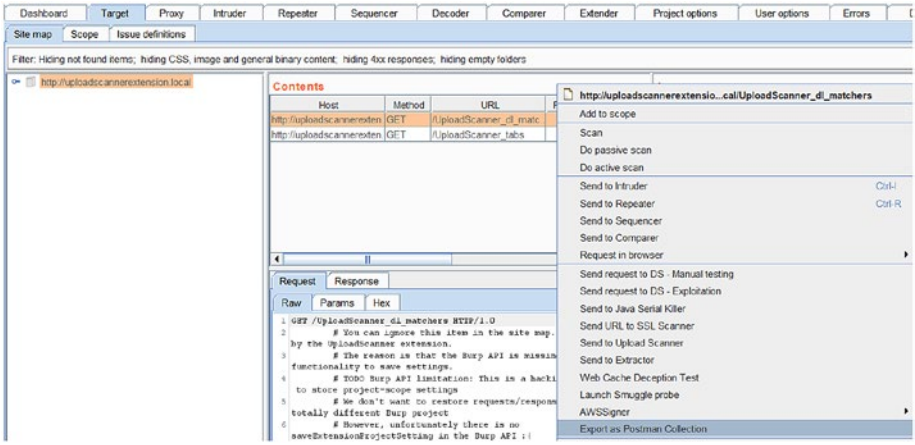

Рис.10-6. *Экспорт запросов в Burp Suite в виде коллекции Postman*

Появится новое окно, как показано на рисунке 10-7. Введите требуемое имя коллекции и имя папки и нажмите на кнопку Export. После этого коллекция будет экспортирована и сохранена в указанном месте.

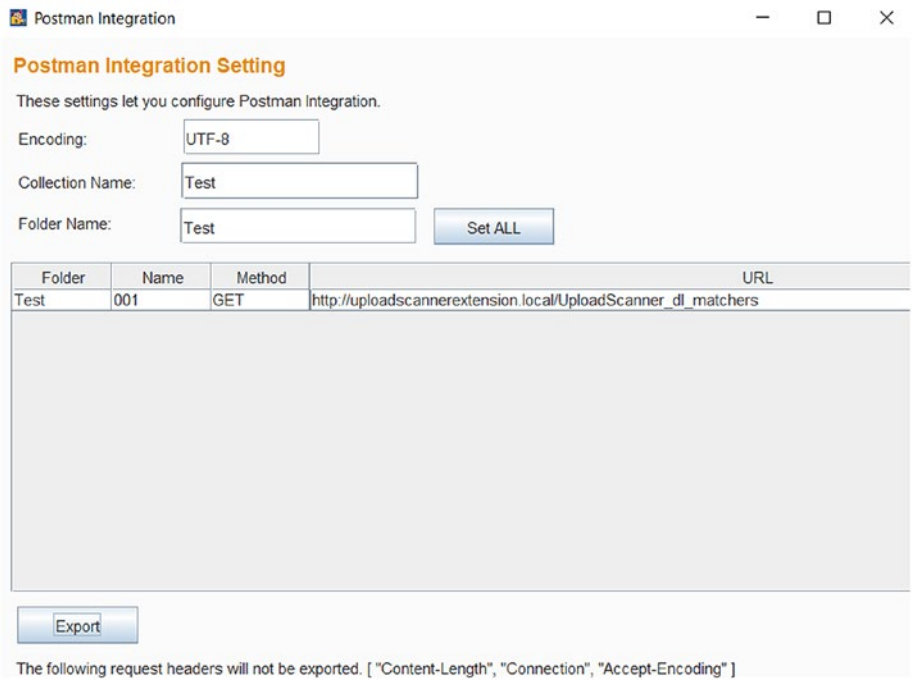

Рис.10-7. *Настройка параметров интеграции Postman*

Затем откройте приложение Postman и нажмите на Import, как показано на рисунке 10-8. Нажмите на выбор файлов и выберите файл, который мы экспортировали из Burp Suite ранее.

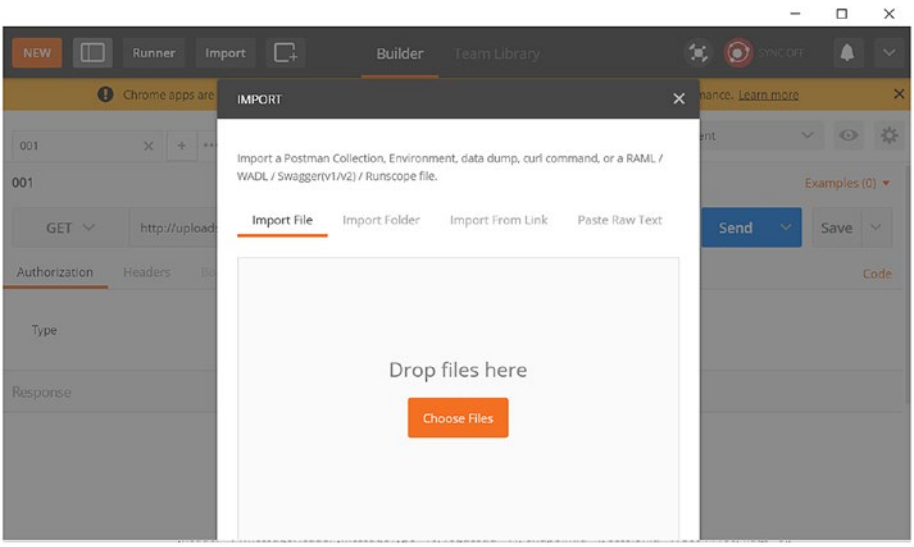

Рис.10-8. *Импорт коллекции в инструмент Postman*

Теперь мы можем увидеть API-запрос, экспортированный из Burp Suite в приложение Postman, как показано на рисунке 10-9.

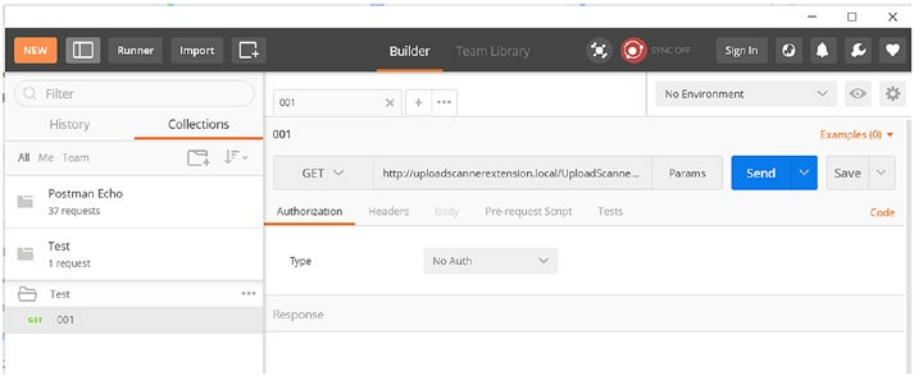

Рис.10-9. *Коллекция, импортированная в инструмент Postman*

Тестирование безопасности мобильных приложений с помощью Burp Suite

В предыдущем разделе мы рассмотрели, как Burp Suite можно настроить вместе с Postman для проведения тестирования безопасности API. В этом разделе мы рассмотрим, как можно использовать возможности Burp Suite для проведения тестирования безопасности мобильных приложений.

Прежде чем мы углубимся в детали тестирования безопасности мобильных приложений, важно понять тот факт, что Burp Suite буквально действует и служит в качестве HTTP-прокси. Это означает, что мы можем эффективно использовать Burp Suite для проведения тестирования безопасности на любом устройстве или приложении, взаимодействующем по протоколу HTTP или HTTPS.

Мобильные приложения ничем не отличаются; они используют тот же протокол HTTP / HTTPS для связи; следовательно, трафик может быть направлен через Burp Suite так же, как и через любое другое обычное веб-приложение. На рисунке 10-10 показан клиент мобильного приложения (эквивалент браузера на ПК), который передает весь трафик через Burp Suite на сервер приложений.

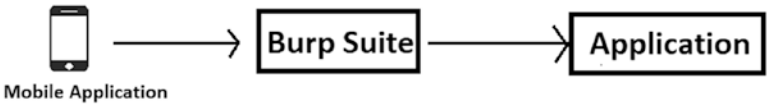

Рис.10-10. *Подключение мобильного приложения к Burp Suite*

Теперь мы посмотрим, как настроить Burp Suite для работы вместе с мобильным приложением. Во-первых, нам нужно убедиться, что правильный прокси Burp Suite настроен на прослушивание на всех интерфейсах. Для этого перейдите на вкладку Proxy > Options, как показано на рисунке 10-11.

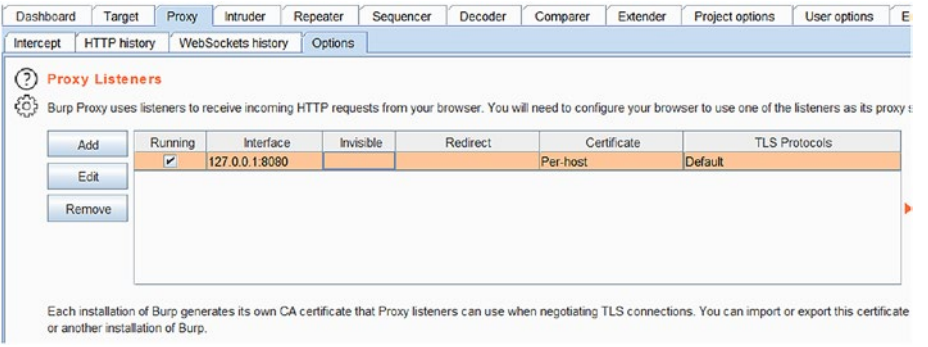

Рис.10-11. *Настройка прокси-прослушивателя в Burp Suite*

Теперь нажмите на кнопку 'Add', появится новое окно, как показано на рисунке 10-12. Настройте номер порта и выберите 'Bind to address' как 'All interfaces' и нажмите 'OK'.

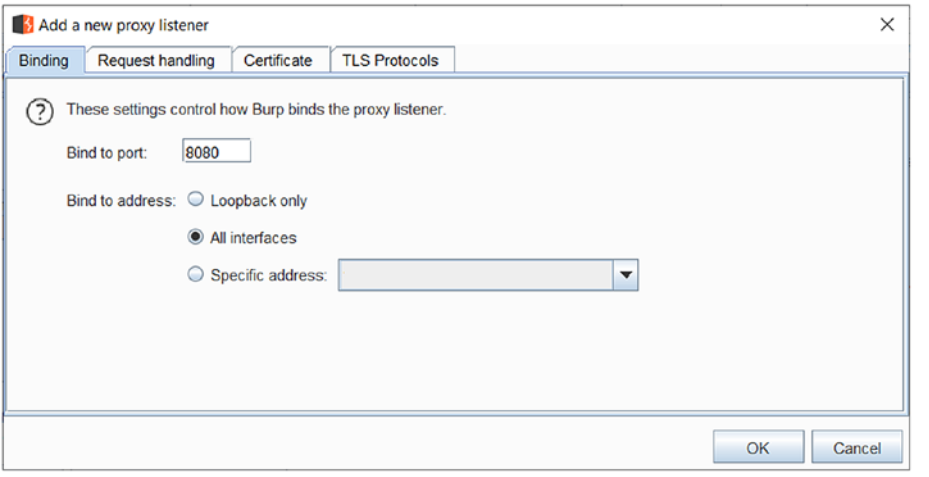

Рис.10-12. *Настройка прокси-прослушивателя в Burp Suite*

Теперь обратите внимание на раздел "proxy listener", как показано на рисунке 10-13, в котором интерфейс указан как '*:8080'.

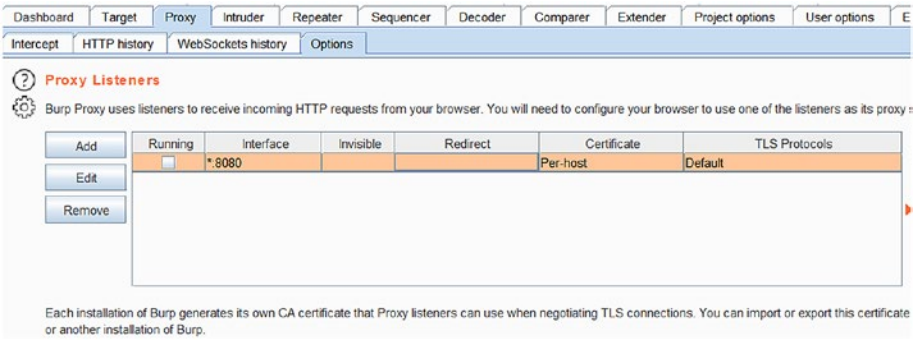

Рис.10-13. *Настройка прокси-прослушивателя в Burp Suite*

Теперь, когда мы настроили прокси Burp Suite на прослушивание порта 8080 на всех доступных интерфейсах, перейдем к настройке мобильного устройства.

Важно отметить, что для настройки мобильного устройства на работу вместе с Burp Suite, система на которой работает Burp Suite, и мобильное устройство должны находиться в одной локальной сети. Самый простой способ добиться этого - подключить мобильное устройство и систему, на которой работает Burp Suite, к одной и той же точке беспроводного доступа. После того как мобильное устройство и система, на которой запущен Burp Suite, подключены к одной сети, нам нужно настроить сетевые параметры на мобильном устройстве для использования Burp Suite в качестве прокси.

Чтобы настроить прокси на мобильном устройстве, перейдите в Настройки беспроводной сети и выберите Беспроводную сеть, к которой вы подключены, как показано на рисунке 10-14.

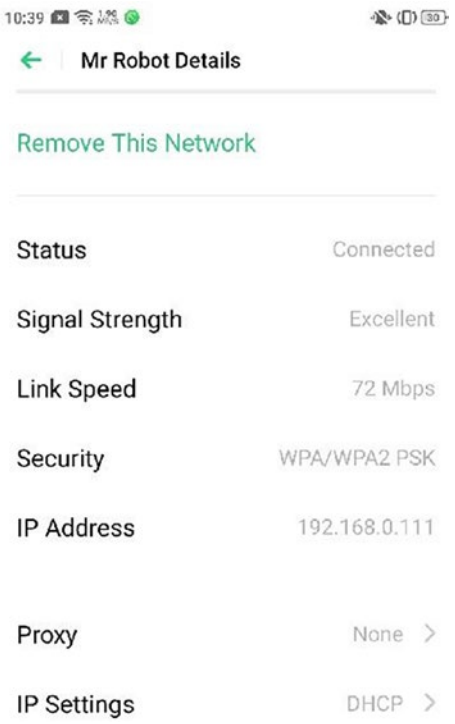

Рис.10-14. *Настройка прокси на мобильном устройстве*

Теперь нажмите на опцию Proxy, и откроется новое окно конфигурации, как показано на рисунке 10-15.

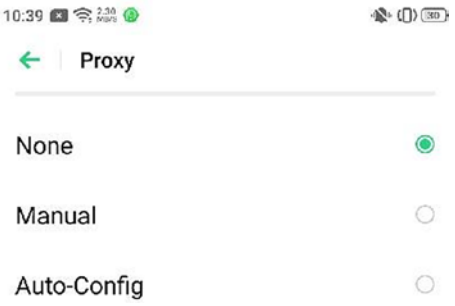

Рис.10-15. *Настройка прокси на мобильном устройстве*

По умолчанию прокси установлен на 'None'. Нам нужно изменить это значение на 'Manual', как показано на рисунке 10-16.

Рис.10-16. *Настройка прокси на мобильном устройстве*

Теперь, когда мы изменили тип прокси на "Manual", нам нужно ввести IP адрес системы, на которой запущен Burp Suite, а также номер порта, на котором прослушивается прокси Burp Suite, как показано на рисунке 10-17.

Рис.10-17. *Настройка прокси на мобильном устройстве*

После настройки "Manual" прокси весь трафик, исходящий от мобильного приложения, будет направляться через Burp Suite. Как только запросы попадают в Burp Suite, их можно подделать с помощью Repeater или Intruder, как и любой другой обычный HTTP-запрос.

Важно отметить два момента в отношении проведения тестирования безопасности мобильных приложений с помощью Burp Suite:

1. Burp Suite может только помочь выполнить ручные тесты безопасности мобильного приложения и, в определенной степени, провести динамическое тестирование безопасности приложения.
2. Точный процесс настройки прокси на мобильных устройствах зависит от типа и версии операционной системы, на которой они работают. Однако на высоком уровне процесс будет похож на то, что мы обсудили в этом разделе.

Рабочий процесс тестирования безопасности с помощью Burp Suite

На протяжении всей книги мы рассмотрели все аспекты Burp Suite и его возможностей. Мы увидели различные инструменты и утилиты, которые предоставляются из коробки, а также использование расширений сторонних производителей, которые значительно расширяют возможности Burp Suite.

Теперь, поскольку мы находимся в конце книги, стоит подвести итоги рабочего процесса или подхода для эффективного использования Burp Suite для тестирования безопасности веб-приложений.

Ниже приведен поэтапный подход, которому можно следовать для эффективного использования Burp Suite:

1. **Правильная установка и конфигурация** - Прежде чем мы начнем использовать Burp Suite, важно, чтобы он был правильно установлен и сконфигурирован.
    a. Убедитесь, что используется нужная редакция и последняя версия Burp Suite.
    b. Настройте параметры прокси в браузере для работы вместе с Burp Suite.
    c. Установите сертификат Burp Suite CA в браузер.

d. Настройте аутентификацию платформы, вышестоящий прокси-сервер и socks-прокси, если это необходимо.

e. Посмотрите и определите горячие клавиши Burp Suite.

f. Убедитесь, что автоматическое резервное копирование проекта включено и настроено правильно.

g. Убедитесь, что такие параметры проекта, как тайм-ауты, разрешение имени хоста, запросы вне зоны действия, перенаправления, конфигурация TLS, правила обработки сеансов, Cookie Jar и макросы настроены в соответствии с требованиями.

h. Убедитесь, что прокси-прослушиватель настроен и работает должным образом.

i. Убедитесь, что все необходимые расширения установлены и загружены.

j. Убедитесь, что целевое приложение может быть исправлено с помощью Burp Suite Infiltrator.

2. **Обход и понимание приложения** - Как только Burp Suite соответствующим образом настроен, важно просмотреть и изучить целевое приложение, чтобы узнать о нем больше.

a. Используйте функцию обхода (Crawl) в сканере для просмотра приложения.

b. Воспользуйтесь функцией обнаружения контента.

c. Вручную просматривайте критические рабочие процессы.

d. Внимательно следите за целевой вкладкой и отслеживайте HTTP-запросы.

e. Найдите и выделите интересные запросы с параметрами.

f. Используйте функцию анализа цели, чтобы получить обзор области применения.

g. Используйте инструменты взаимодействия для поиска комментариев, сценариев и ссылок.

h. Отслеживайте проблемы, о которых сообщают пассивные сканеры.

3. **Атакуйте приложение** - Теперь, когда мы провели достаточно разведки, пришло время атаковать выборочные функциональные возможности приложения.

   a. Запустите задачу аудита с помощью сканера Burp Suite.

   b. Найдите интересные запросы, особенно те, которые содержат параметры, и отправьте запрос в Repeater для дальнейшего изучения.

   c. Вносите изменения в запрос, параметры, заголовки и тело с помощью repeater.

   d. Если запрос и параметр необходимо проверить на массовую полезную нагрузку, воспользуйтесь Intruder.

   e. Используйте comparer для анализа и интерпретации различий в различных ответах.

   f. Используйте sequencer для проверки прочности токенов.

   g. Используйте decoder для кодирования или декодирования любого из значений параметров по мере необходимости.

   h. Проверьте наличие уязвимостей Clickjacking с помощью инструмента Clickbandit.

   i. Сгенерируйте доказательство концепции (PoC) для атаки Cross-Site Request Forgery с помощью CSRF PoC generator.

   j. Используйте collaborator в Burp Suite для эффективного обнаружения внеполосных уязвимостей, таких как XML External Entity Injection (XXE) и Server Side Request Forgery (SSRF).

k. Отслеживайте все найденные уязвимости на панели "issues" во вкладке "target".

l. Выберите необходимые уязвимости и экспортируйте их в HTML-отчет.

Резюме

В этой главе мы рассмотрели, как можно использовать возможности Burp Suite для проведения тестирования безопасности API, а также мобильных приложений. Мы также кратко описали рабочий процесс, которому можно следовать, чтобы наилучшим образом использовать Burp Suite для тестирования безопасности веб-приложений.

Упражнения

1. Настройте Postman для работы вместе с Burp Suite. Перехватите API-запросы в Burp Suite и атакуйте их с помощью Repeater.
2. Протестируйте любое из целевых приложений, используя полный рабочий процесс, рассмотренный в этой главе.

Индекс

S. Rahalkar, *A Complete Guide to Burp Suite*,
https://doi.org/10.1007/978-1-4842-6402-7