Web Security Basics – Exercise

By Sebastian Medina

Contents

1.1 DNS in detail

Summary:

In this module I learned what DNS is and how it is used in the real world. DNS stands for Domain Name System, and it is a simple way for people to communicate with devices. So, they do not have to remember the websites Ip address they can just use remember something simple like tryhackme.com. I then learned about the Domain Hierarchy; it starts with the Root Domain also known as the main part of your website domain name. Then you move to TLD (Top-Level Domain), this is the most righthand part of the domain name some examples are .com, .gov, and .edu. Within the TLD you have two types of TLD you have gTLD (Generic Top Level) and ccTLD (Country Code Top Level Domain). gTLD tells the user the domain name's purpose like .edu is for education and .gov is for the government. ccTLD is used for geographical purpose like .ca is a site based out of Canada. Back to the Hierarchy you then get the Second-Level Domain, this domain is the middle part of a domain. Take tryhackme.com the Second-Level Domain is tryhackme. In this category you can only have 63 characters. Lastly you get the Subdomain which is on the left-hand side of the Second-Level Domain. You can have multiple subdomains however each subdomain can only be 63 characters long. After talking about the domain hierarchy, I learned the different types of DNS records; There is the A record that resolves the IPV4 addresses, then you have the AAAA Record that resolves the IPv6 addresses, there is the CNAME Record that resolves to another domain name, there is the MX Records that resolves to the address of the servers that handle the email for the domain you are querying, and lastly there is the TXT Record they are free text field where any text-based data can be stored. I then learned about how DNS makes a request from you to the DNS authoritative DNS Server however there are many places it can stop because the request was meet like Recursive DNS Server if your request has something to do with the local cache it has within it.

DNS in detail

Learn how DNS works and how it helps you access internet services.

DNS in Detail • Jul 10, 2021

DNS in Detail - How the web works

Task 1 What is DNS?

Task 2 Domain Hierarchy

Task 3 Record Types

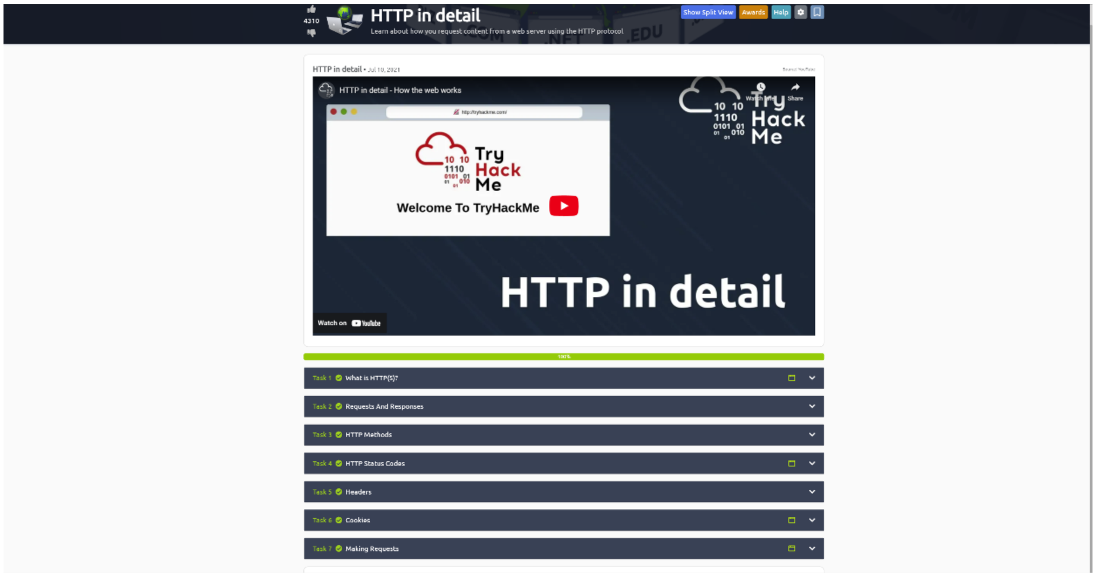
Task 4 Making A Request

Task 5 Practical

1.2 HTTP in detail

Summary:

Within this module I learned what HTTP is and that it stands for HyperText Transfer Protocol. HTTP is used whenever you view a website. I also learned what HTTPS (HyperText Transfer Protocol Secure) is just a secure version of HTTP. Then the module taught me abourt request and responses, this happens when ever you access a website your browser needs to make a request to the Web server for HTML code, Images, and any other content within the webserver then the webserver downloads the responses. Some HTTP Methods that I learned were GET request that is used for getting information from a web server, Post Request used for submitting data, PUT request that Submits data, and Delete request that deletes information and records. After this I learned about status codes and what a certain range of numbers most likely means. For example, if I see anything between 100-199 it is most likely an Information response. Headers in HTTP are not required by any means however it does make viewing the website much better. Lastly the module taught me about cookies, they are a small piece of data that is stored on your device and can help the webserver who you are and what some of your personal settings are.



HTTP in detail

Learn about how you request content from a web server using the HTTP protocol

HTTP in detail • Jul 10, 2021

HTTP in detail - How the web works

Welcome To TryHackMe

HTTP in detail

Task 1 What is HTTP(S)?

Task 2 Requests And Responses

Task 3 HTTP Methods

Task 4 HTTP Status Codes

Task 5 Headers

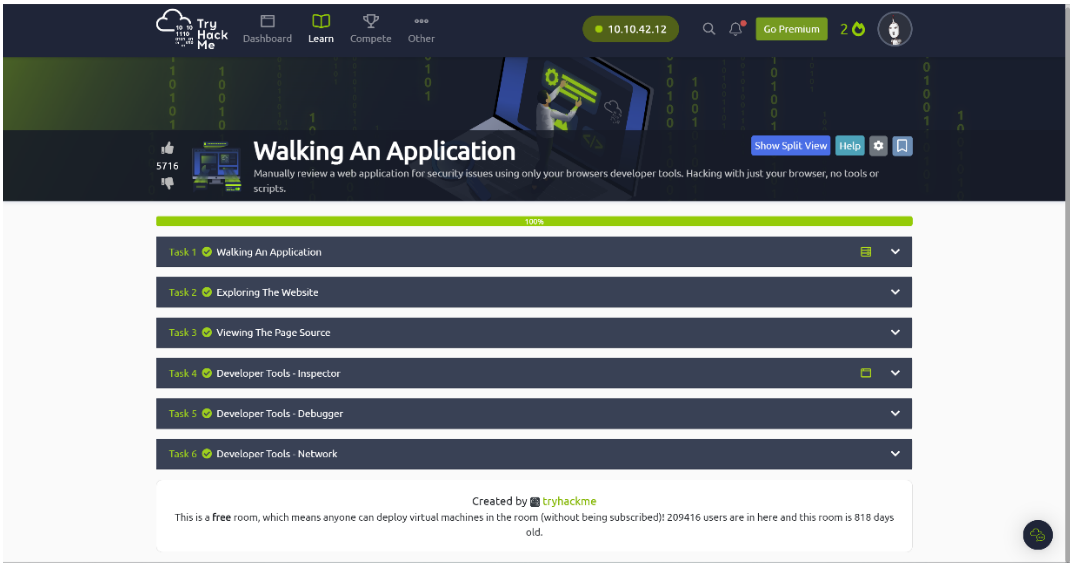
Task 6 Cookies

Task 7 Making Requests

2.1 Walking An Application

Summary:

In the Walking An Application module I learned how to manually review a web application for security issues by only using the built in tools within my own browser. This is the job of a penetration tester to see if certain features can put the website in danger of being hacked. Some of the tools that were taught in this module are the View page source, Developer Tool-Inspector, Developer Tool - Debugger, and Developer Tool – Network. The view page source allows you to see human readable code that is returned to the client from the webserver every time you make a request. The Inspector tool allows you to see even things that are not on the webpage like comments. The Debugger tool allows us to dig deeper into the JavaScript code and see if there is any vulnerability. The last tool I learned in this module is the network tool that allows you to keep track of every external request a webpage makes.



Walking An Application

Manually review a web application for security issues using only your browsers developer tools. Hacking with just your browser, no tools or scripts.

Task 1 ✓ Walking An Application

Task 2 ✓ Exploring The Website

Task 3 ✓ Viewing The Page Source

Task 4 ✓ Developer Tools - Inspector

Task 5 ✓ Developer Tools - Debugger

Task 6 ✓ Developer Tools - Network

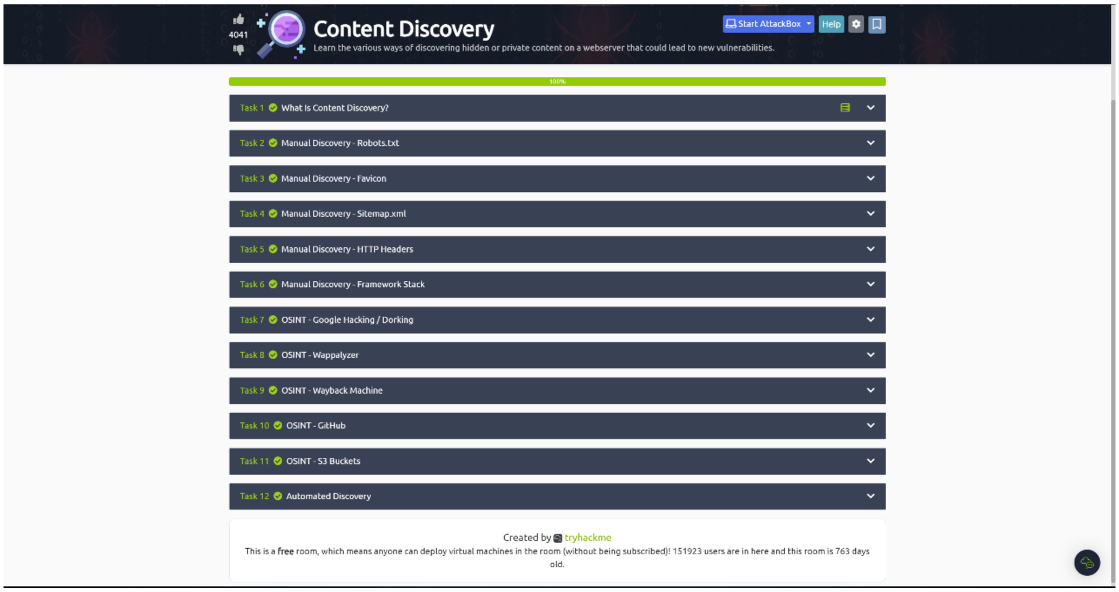
Created by tryhackme

This is a **free** room, which means anyone can deploy virtual machines in the room (without being subscribed)! 209416 users are in here and this room is 818 days old.

2.2 Content Discovery

Summary:

Within the Content Discovery module, I learned the three main ways of discovering content on a website. There is manual, automated and OSINT. This module first talks about manual discovery like Favicon, Sitemap.xml, HTTP headers, and Framework Stack. Favion is a small icon on the website address bar that brand the website this also falls into the Framework Stack and allows for a website to have copyrights. Then there is Sitemap.xml that gives a list of every file the website owner wishes to be listed on a search engine. HTTP headers can contain useful information like programming language or even webserver software. Then the module moves to OSINT Discovery and uses Google Hacking, Wappalyxer, Wayback Machine, GitHub, and S3 Buckets. Google hacking allows you to pick out custom content so you can control what you want to see. Then I learned about Wappalyxer and this is a an online tool that is able to identify what technologies a website is using like the framework all the way to Content Management. With Wayback Machine you can use a historical archive website and whenever you search up a domain name it will show you all the time the service saved the webpage contents. Then there is GitHub and S3 Buckets, GitHub can track and change files in a project and S3 Buckets are storage services provided by Amazon AWS. After learning about OSINT the module teaches you about automated discovery. Automated discovery is using tools to discover content rather than doing it manually. This process usually contain hundreds of requests to a web server.



Content Discovery

Learn the various ways of discovering hidden or private content on a webserver that could lead to new vulnerabilities.

Task 1 ✓ What Is Content Discovery?

Task 2 ✓ Manual Discovery - Robots.txt

Task 3 ✓ Manual Discovery - Favicon

Task 4 ✓ Manual Discovery - Sitemap.xml

Task 5 ✓ Manual Discovery - HTTP Headers

Task 6 ✓ Manual Discovery - Framework Stack

Task 7 ✓ OSINT - Google Hacking / Dorking

Task 8 ✓ OSINT - Wappalyzer

Task 9 ✓ OSINT - Wayback Machine

Task 10 ✓ OSINT - GitHub

Task 11 ✓ OSINT - S3 Buckets

Task 12 ✓ Automated Discovery

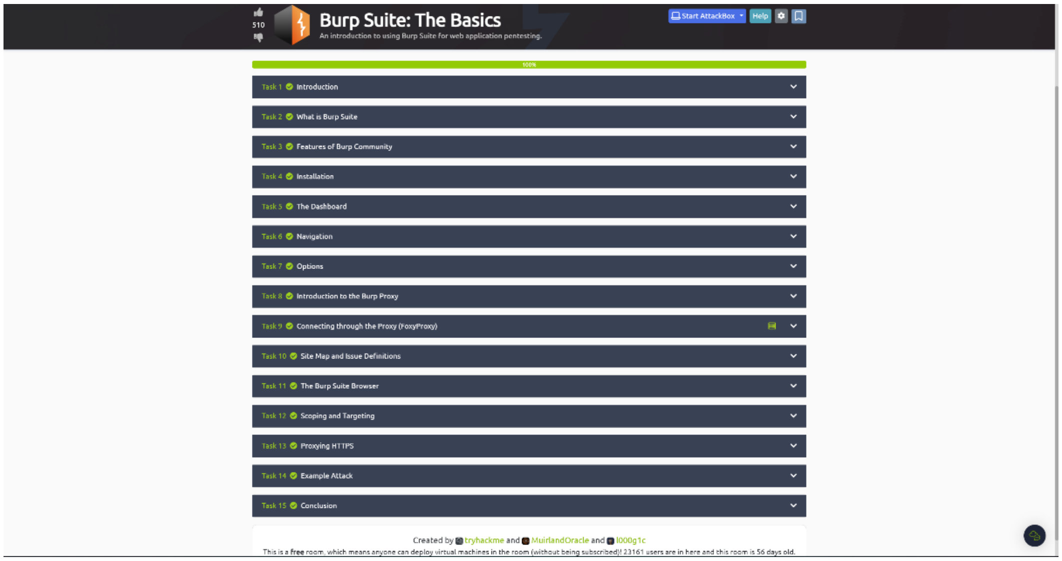
Created by tryhackme

This is a free room, which means anyone can deploy virtual machines in the room (without being subscribed)! 151923 users are in here and this room is 763 days old.

3.1 Burp Suite: The Basics

Summary:

Within the Burp Suite: The Basics I learned the Basics of Burp Suite web application security testing framework. I also learned about the different tools that are available within the framework. A Detail guide on how to install Burp Suite on to my System and how to navigate and configure Burp Suite. Burp Suite is a Java based framework to conduct web applications penetration testing. Some features of Burp Suite are, Proxy: allows interception of request and response, Repeater enables you to capture and resend the same request multiple times, Intruder: can be used for brute-force attacks, Decoder: decodes captured information or can encode payloads, Sequencer: is used when assessing the randomness of tokens and Comparer: this compares two pieces of data. The next couple of tasks in this module is about installing Burp Suit and learning where all the features are located. Then the module takes a deep dive on Burp Proxy. Burp Proxy is a crucial tool within Burp Suite. With the captured information you gathered with proxy you can manipulate and monitor all traffic that you capture in this way.



Burp Suite: The Basics

An introduction to using Burp Suite for web application pentesting.

Task 1 ✓ Introduction

Task 2 ✓ What is Burp Suite

Task 3 ✓ Features of Burp Community

Task 4 ✓ Installation

Task 5 ✓ The Dashboard

Task 6 ✓ Navigation

Task 7 ✓ Options

Task 8 ✓ Introduction to the Burp Proxy

Task 9 ✓ Connecting through the Proxy (FoxyProxy)

Task 10 ✓ Site Map and Issue Definitions

Task 11 ✓ The Burp Suite Browser

Task 12 ✓ Scoping and Targeting

Task 13 ✓ Proxying HTTPS

Task 14 ✓ Example Attack

Task 15 ✓ Conclusion

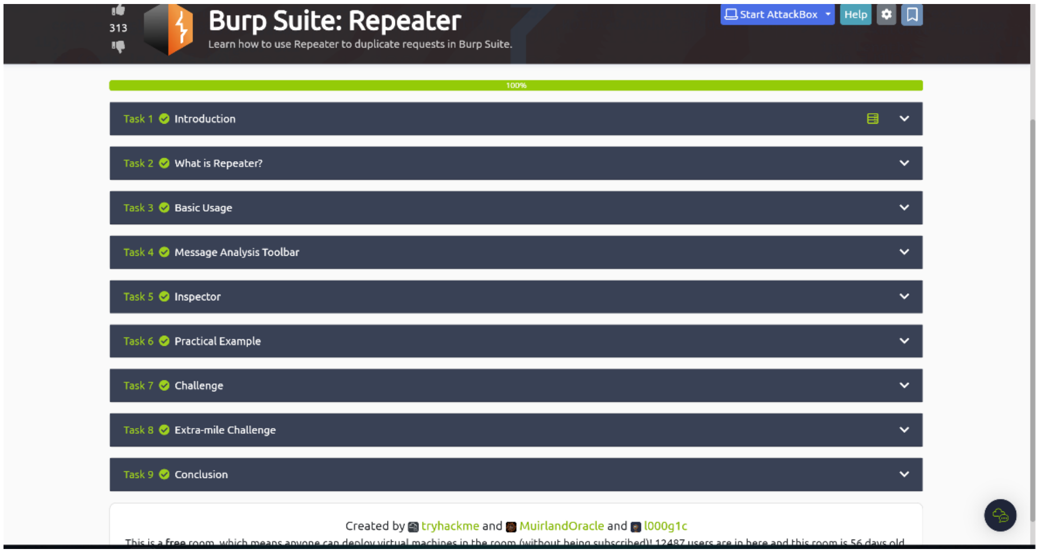
Created by tryhackme and MuirlandOracle and l000g1c

This is a free room, which means anyone can deploy virtual machines in the room (without being subscribed)! 23161 users are in here and this room is 56 days old.

3.2 Burp Suite: Repeater

Summary:

Within this module the focus is Burp Suite Repeater and how the tool can manipulate and resend captures request. Burp Proxy and Burp Repeater go hand in hand because Burp Proxy captures the request and then manipulates it and sends the modified request as many times as you need. Then the module walks the user though the Repeater tool within Burp Suite that was just recently installed on the user computer. The module shows you how to use it and how to complete certain tasks with Repeater within Burp Suite. After this module I feel like I have learned a lot about repeater because most of this Module is hands on and you have to navigate Burp Repeater in order to complete the task within the module.



Burp Suite: Repeater

Learn how to use Repeater to duplicate requests in Burp Suite.

Task 1 ✓ Introduction

Task 2 ✓ What is Repeater?

Task 3 ✓ Basic Usage

Task 4 ✓ Message Analysis Toolbar

Task 5 ✓ Inspector

Task 6 ✓ Practical Example

Task 7 ✓ Challenge

Task 8 ✓ Extra-mile Challenge

Task 9 ✓ Conclusion

Created by tryhackme and MuirlandOracle and l000g1c

This is a free room, which means anyone can deploy virtual machines in the room (without being subscribed)! 12487 users are in here and this room is 56 days old.