Introduction to Web Attacks

As web applications are becoming very common and being utilized for most businesses, the importance of protecting them against malicious attacks also becomes more critical. As modern web applications become more complex and advanced, so do the types of attacks utilized against them. This leads to a vast attack surface for most businesses today, which is why web attacks are the most common types of attacks against companies. Protecting web applications is becoming one of the top priorities for any IT department.

Attacking external-facing web applications may result in compromise of the businesses' internal network, which may eventually lead to stolen assets or disrupted services. It may potentially cause a financial disaster for the company. Even if a company has no external facing web applications, they likely utilize internal web applications, or external facing API endpoints, both of which are vulnerable to the same types of attacks and can be leveraged to achieve the same goals.

While other HTB Academy modules covered various topics about web applications and various types of web exploitation techniques, in this module, we will cover three other web attacks that can be found in any web application, which may lead to compromise. We will discuss how to detect, exploit, and prevent each of these three attacks.

Web Attacks

HTTP Verb Tampering

The first web attack discussed in this module is HTTP Verb Tampering. An HTTP Verb Tampering attack exploits web servers that accept many HTTP verbs and methods. This can be exploited by sending malicious requests using unexpected methods, which may lead to bypassing the web application's authorization mechanism or even bypassing its security controls against other web attacks. HTTP Verb Tampering attacks are one of many other HTTP attacks that can be used to exploit web server configurations by sending malicious HTTP requests.

Insecure Direct Object References (IDOR)

The second attack discussed in this module is Insecure Direct Object References (IDOR). IDOR is among the most common web vulnerabilities and can lead to accessing data that should not be accessible by attackers. What makes this attack very common is essentially the lack of a solid access control system on the back-end. As web applications store users' files and information, they may use sequential numbers or user IDs to identify each item. Suppose the web application lacks a robust access control mechanism and exposes direct references to files and resources. In that case, we may access other users' files and information by simply guessing or calculating their file IDs.

XML External Entity (XXE) Injection

The third and final web attack we will discuss is XML External Entity (XXE) Injection. Many web applications process XML data as part of their functionality. Suppose a web application utilizes outdated XML libraries to parse and process XML input data from the front-end user. In that case, it may be possible to send malicious XML data to disclose local files stored on the back-end server. These files may be configuration files that may contain sensitive information like passwords or even the source code of the web application, which would enable us to perform a Whitebox Penetration Test on the web application to identify more vulnerabilities. XXE attacks can even be leveraged to steal the hosting server's credentials, which would compromise the entire server and allow for remote code execution.

Let's get started by discussing the first of these attacks in the next section.

Next ➜ Mark Complete & Next

Cheat Sheet

Table of Contents

My Workstation

OFFLINE

Start Instance

∞ / 1 spawns left