Forced Browsing Attack

https://campus.barracuda.com/doc/42049348/

Description

Forced Browsing is an attack technique used to gain access to restricted pages or other sensitive resources in a web server by forcing the URL directly. If the restricted URLs, scripts, or files that reside in the web server directory are not enforced with appropriate authorization, they can be vulnerable to forced browsing attacks.

Attackers typically try brute-force attempts to enumerate directories and files that are restricted from public viewing. Typically, the files/directory paths have common naming conventions, and can therefore be easily guessed using brute force. The brute-force attack is manually executed if the directories/pages are based on predictable resources, or use automated tools for common files and directories. Predictable resources can also be guessed by analyzing the HTTP response code of the web server.

Forced browsing is also known as Forceful Browsing, File Enumeration, Predictable Resource Location, and Directory Enumeration.

Effects

If a web server or a web application is vulnerable to forced browsing attacks, an attacker can access restricted files and view sensitive information.

Methods

The attack can be done either manually or by using automated tools for common files and directory names. When done manually, an attacker can also predict unlinked resources when URLs are generated in a predictable manner or use number rotation techniques. Most commercial and open-source scanners also typically scan for predictable resources that are not directly linked but contain sensitive information.

Examples

Forced browsing attempted on the website that did not enforce proper checks before processing any operation:

In this example, an attacker directly accesses a URL which performs certain operations on the server without following the workflow of the web application.

Consider an online money transfer option that is available on a bank website, which allows a web user to login to his account and transfer money. Analyzing these HTTP requests for the online money transfer, the attacker might find that the URL for this operation is `http://www.ABC-bank.com/transfer-money.asp?From_account=123456&To_account=7891011&amount=10000`. If the attacker uses this URL and tries a different `From_account` value with their own `To_account` number, they might succeed in transferring money to account without the consent of the `from` account holder.

If the web application does not verify that the first step was performed successfully (checking that the user is logged into the account), before performing the second step (performing a money transfer to any other account), it provides an opening for an attacker to perform forced browsing.

Forced browsing attempted on a web server that did not enforce authorization for restricted files:

If a web server (www.example.com) has a page like `admin.asp`, `admin.jsp`, or `admin.php` that contains sensitive information related to the server, the page might not be accessible by the normal web user. If the attacker wants to access the `admin.asp` or `admin.jsp` page to steal sensitive information, the attacker might try the following:

- http://www.example.com/admin.asp
- http://www.example.com/admin.jsp

You can see how the attacker can perform forced browsing to access sensitive pages from the server. If the proper permissions or ACLs are configured for these pages, the attacker will not be able to access these kinds of files.

Prevention

There are two ways to protect against forced browsing – enforcing an application URL space allow list and using proper access control.

Creating an allow list (or whitelist) involves allowing explicit access to a set of URLs that are considered to be a part of the application to exercise its functionality as intended. Any request not in this URL space is denied by default. Manually creating and maintaining such a list can be tedious. You can use the Barracuda Web Application Firewall's adaptive profiling to automatically create such a list and enforce it by learning the valid URL space from trusted traffic. It also comes with a block list of

common files and directories that are commonly left exposed unintentionally.

In the second method, using proper access control and authorization policies, access is only given to users commensurate with their privileges. The Barracuda Web Application Firewall provides authorization policies at a URL level along with protection against session-based attacks to provide proper access control enforcement against such abuse.

Tags

OWASP Top 10, PCI-DSS

See Also

CWE 425, CPEC-87, OWASP, WASC