Testing static websites and uncovering hidden security vulnerabilities

BY INTIGRITI · MARCH 14, 2024 · LAST UPDATED ON MARCH 6, 2025

By not conducting tests on the static websites of your targets, you may be overlooking numerous potential vulnerabilities. In today's post, we will go through the top 3 most common ways of finding security vulnerabilities in static websites.

What are static websites?

You've probably come across a static site before. They often reject user input, making them largely impervious to injection attacks.

There are various forms and uses of them, including blogs generated through static means, documentation for products or APIs, and occasionally servers that distribute content such as JavaScript files and images.

Top 3 most common vulnerabilities found in static sites

1) Exposed backup files

One of the most frequently encountered security flaws in static websites is the presence of accessible backup files that can be downloaded by anyone. At times, they are automatically created by an automated system and mistakenly placed in the web root directory, thus granting access to anyone who can locate the file.

There are multiple methods for verifying potential paths or areas that could lead you to locating a backup file. A popular method for achieving this is by performing a brute-force attack on hidden directories and files. Let's go through how this would work in practice.

First, grab or generate a custom wordlist with specific backup filenames. Next up, we can use a brute-forcing fuzzer tool, like FFuF (we have documented this tool in one of our previous blog posts). The final step requires you to run the tool against your target that you want to scan.

Besides forced directory browsing, there are also several other (passive) methods that you can use to enumerate links or files. For example, you could try to look for backup files:

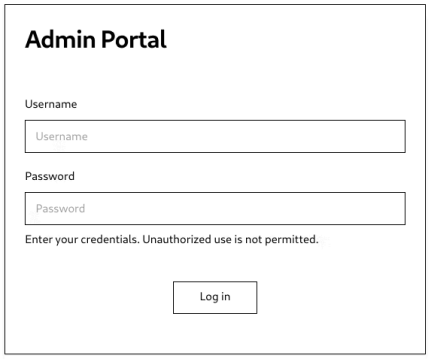
- On common server config files like (Robots.txt and Sitemap.xml)
- By enumerate files using any directory listings (often caused by a misconfigured server and a missing index file)

- On any potential disclosed source code and JavaScript files as these often contain comments with references
- Through any search or indexing engine like Google or Bing and Wayback Machine.

2) Exposed portals and panels

Another commonly found issue is portals or panels that are left open for any external visitor to access. These portals are often used by the site owners to manage and/or bring changes to the content served (think about a blog that has a portal for publishers to write and publish new posts).

Its safe to say that most come with elevated privileges and that you should always keep an eye on any unprotected portals. Just as before, you can enumerate portals or panels using several content discovery methods through:



- Common server configuration files (such as the Robots.txt file and Sitemap.xml)
- Any disclosed source code or JavaScript files (especially comments)
- Search engines like Google, Bing and DuckDuckGo
- Internet Archives like Wayback Machine

3) Vulnerable web servers

A lot of bug bounty hunters do not pay attention to this small single detail that gets sent in the response header: **the web server's version**.

```
HTTP/2 200 OK
Date: Thu, 14 Mar 2024 13:37:07 GMT
Content-Type: text/html; charset=utf-8
Server: Apache 2.3.10
Content-Control: private

...
```

There is a strong chance that the server has not been updated automatically on a regular basis, or that any updates have been done manually with long intervals in between, resulting in a potentially outdated server.

You should always make sure to check the version number for any matching publicly released exploits. ExploitDB is one of the most commonly used source for publicly released exploits.

In conclusion

These were the top 3 most common security vulnerabilities found on static sites! We hope that you've learned something new from this post!

Looking to hunt on some targets that feature static sites and to try out your new skillset? Browse through our 70+ public bug bounty programs on Intigriti and who knows, maybe you'll earn a bounty with us!

START HUNTING ON INTIGRITI TODAY