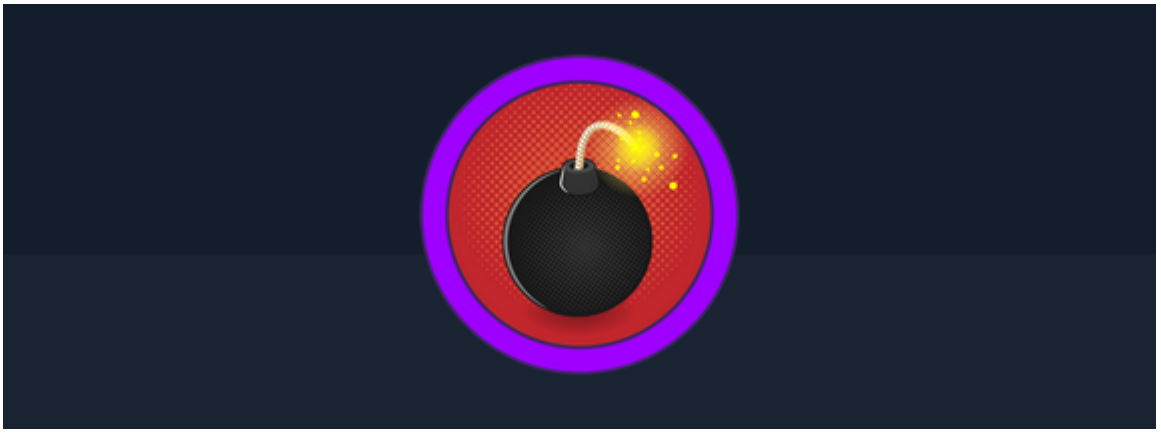
Ignition

Ignition HTB

Cover



Target: HTB Machine “Ignition” **Client:** Megacorp (Fictitious) **Engagement Date:** May 2025 **Report Version:** 1.0

Prepared by: Jonas Fernandez

1. Introduction

Objective of the Penetration Test

The primary objective of this penetration testing engagement was to identify security weaknesses within a Linux-based target system hosted at **10.129.71.172** (accessible via the domain ignition.htb). Our evaluation focused on uncovering potential misconfigurations in the web application's Magento framework, particularly vulnerabilities in its administrative interface that allowed unauthorized access through weak authentication measures. The goal was to expose these critical issues and provide actionable recommendations to enhance the overall security posture of the system.

Systems Evaluated

The assessment centered on the publicly accessible web service running on port **80**, which is powered by the Magento e-commerce platform. Our evaluation included:

- **Magento Administrative Interface:** A comprehensive review of the Magento admin portal, where we identified that a weak password policy allowed access using commonly guessed credentials.

Legal and Ethical Considerations

This penetration test was conducted with explicit authorization from the designated authority and in strict adherence to ethical guidelines and industry best practices. All activities were performed in a manner that ensured the normal operations of the target system were not disrupted. The findings detailed within this report are confidential and intended solely for internal stakeholders to support remediation efforts.

2 Methodology

1. Initial Host Discovery and OS Fingerprinting

- **Objective:** Verify that the target system is online and determine its operating system.
- **Procedure:** Executed a ping command:

```
kali@kali ~/workspace [15:28:53] $ ping -c 1 10.129.71.172
PING 10.129.71.172 (10.129.71.172) 56(84) bytes of data.
64 bytes from 10.129.71.172: icmp_seq=1 ttl=63 time=54.8 ms
```

```
--- 10.129.71.172 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 54.826/54.826/54.826/0.000 ms
```

- **Observation:** The response included a TTL value of 63, indicating that the target is likely running a Linux-based operating system.

2. Port Scanning

- **Objective:** Identify open ports on the target system.
- **Procedure:** Performed a full TCP SYN scan using Nmap:

```
kali@kali ~/workspace [15:30:02] $ sudo nmap -sS -p- --open -n -Pn
10.129.71.172 -oG Ignitionports
[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-30 15:30 EDT
Nmap scan report for 10.129.71.172
Host is up (0.040s latency).
Not shown: 65534 closed tcp ports (reset)
PORT   STATE SERVICE
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 16.60 seconds
```

- **Observation:** The scan revealed that only port **80/tcp** is open, confirming that the host is running a web service.

3. Service Enumeration

- **Objective:** Gather detailed information about the web service running on port 80.
- **Procedure:** Ran Nmap with version detection and included default scripts:

```
kali@kali ~/workspace [15:30:33] $ sudo nmap -sVC -p 80 10.129.71.172 -oN
services
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-30 15:31 EDT
Nmap scan report for 10.129.71.172
Host is up (0.054s latency).

PORT   STATE SERVICE VERSION
80/tcp open  http    nginx 1.14.2
|_http-server-header: nginx/1.14.2
```

```
|_http-title: Did not follow redirect to http://ignition.htb/

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.34 seconds
```

Observation: The output showed that the HTTP service is powered by **nginx 1.14.2**. Additionally, the title indicated a redirection to `http://ignition.htb/`, suggesting that domain-level configurations are in place.

4. Framework Detection

- **Objective:** Identify the underlying web application framework.
- **Procedure:** Used the `whatweb` tool on the target domain:

```
kali@kali ~/workspace [15:34:12] $ whatweb ignition.htb
http://ignition.htb [200 OK] Country[RESERVED][ZZ], HTML5,
HTTPServer[nginx/1.14.2], IP[10.129.71.172], Magento,
Script[text&#x2F;javascript,text/javascript,text/x-magento-init],
Title[Home page], UncommonHeaders[content-security-policy-report-only,x-
content-type-options], X-Frame-Options[SAMEORIGIN], X-XSS-Protection[1;
mode=block], nginx[1.14.2]
```

Observation: The results pointed to **Magento** as the underlying e-commerce framework, along with supporting details like HTML5, various scripting tags, and relevant HTTP headers.

5. Directory and Endpoint Discovery

- **Objective:** Enumerate accessible directories and discover hidden endpoints.
- **Procedure:** Leveraged **ffuf**, a directory brute-forcing tool, with a common wordlist:

```
kali@kali ~/workspace [15:34:42] $ ffuf -w
/usr/share/wordlists/seclists/Discovery/Web-Content/common.txt  -u
http://ignition.htb/FUZZ/


        /'___\  /'___\           /'___\
       /\ \__/ /\ \__/  __  __  /\ \__/
       \ \ ,__\\ \ ,__\/\ \/\ \ \ \ ,__\
        \ \ \_/ \ \ \_/\ \ \_\ \ \ \ \_/
         \ \_\   \ \_\  \ \____/  \ \_\
          \/_/    \/_/   \/___/    \/_/
```

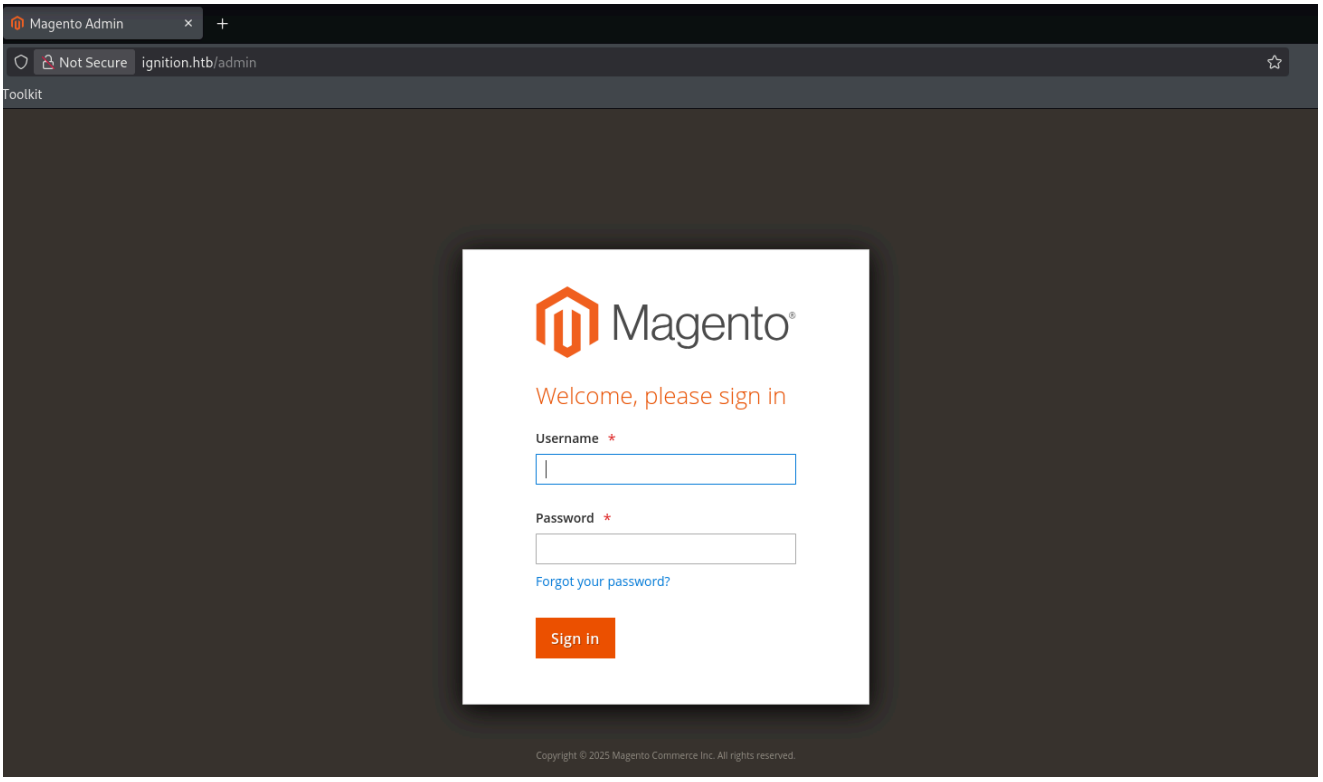
```
       v2.1.0-dev
________________________________________________

 :: Method           : GET
 :: URL              : http://ignition.htb/FUZZ/
 :: Wordlist         : FUZZ: /usr/share/wordlists/seclists/Discovery/Web-
Content/common.txt
 :: Follow redirects : false
 :: Calibration      : false
 :: Timeout          : 10
 :: Threads          : 40
 :: Matcher          : Response status: 200-
299,301,302,307,401,403,405,500
________________________________________________

0                       [Status: 200, Size: 25821, Words: 5441, Lines:
426, Duration: 3115ms]
Home                    [Status: 301, Size: 0, Words: 1, Lines: 1,
Duration: 6253ms]
admin                   [Status: 200, Size: 7095, Words: 1551, Lines: 149,
Duration: 6101ms]
catalog                 [Status: 302, Size: 0, Words: 1, Lines: 1,
Duration: 6085ms]
checkout                [Status: 302, Size: 0, Words: 1, Lines: 1,
Duration: 7941ms]
cms                     [Status: 200, Size: 25815, Words: 5441, Lines:
426, Duration: 5743ms]
contact                 [Status: 200, Size: 28691, Words: 6592, Lines:
504, Duration: 7777ms]
enable-cookies          [Status: 301, Size: 0, Words: 1, Lines: 1,
Duration: 6108ms]
home                    [Status: 301, Size: 0, Words: 1, Lines: 1,
Duration: 5064ms]
index.php               [Status: 200, Size: 25817, Words: 5441, Lines:
426, Duration: 5686ms]
robots.txt              [Status: 200, Size: 1, Words: 1, Lines: 2,
Duration: 7271ms]
robots                  [Status: 200, Size: 1, Words: 1, Lines: 2,
Duration: 7300ms]
setup                   [Status: 200, Size: 2827, Words: 460, Lines: 71,
Duration: 6450ms]
soap                    [Status: 200, Size: 391, Words: 94, Lines: 14,
```

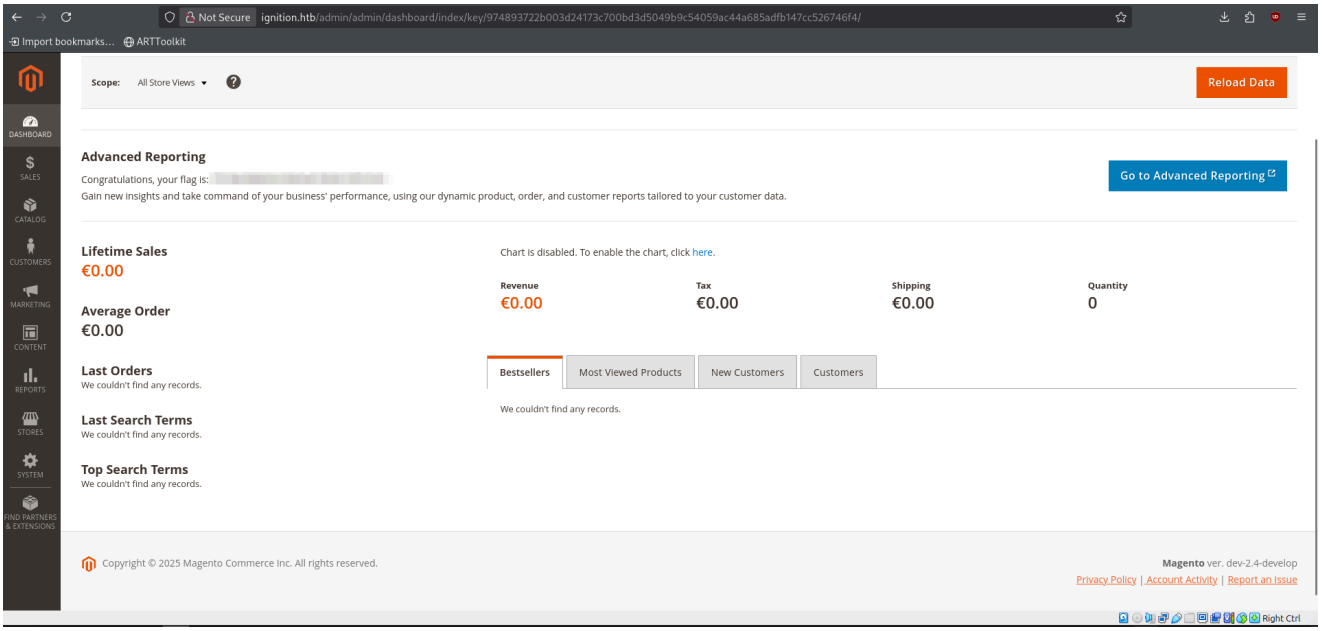
```
Duration: 5969ms]
wishlist                [Status: 302, Size: 0, Words: 1, Lines: 1,
Duration: 5649ms]
```

Observation: The scan uncovered several endpoints—including `/admin`, `/catalog`, `/checkout`, `/cms`, `/contact`, `/robots.txt`, `/setup`, `/soap`, and `/wishlist`. Notably, the discovery of the `/admin` endpoint pointed to the Magento administration interface.

6. Password Policy Analysis and Authentication Testing

- **Objective:** Assess the password policy for the Magento admin interface and test for weak default credentials.
- **Procedure:** After reviewing the Magento admin login page, it was determined that the password policy required a minimum of 7 characters with a combination of letters and numbers (without a strict distinction between uppercase and lowercase). By researching the top 10 most common passwords of 2023, the password `REDACTED` was tested.
- **Observation:** The use of `REDACTED` successfully authenticated, granting access to the Magento admin panel. This successful login confirmed the exploitation of weak password policies.

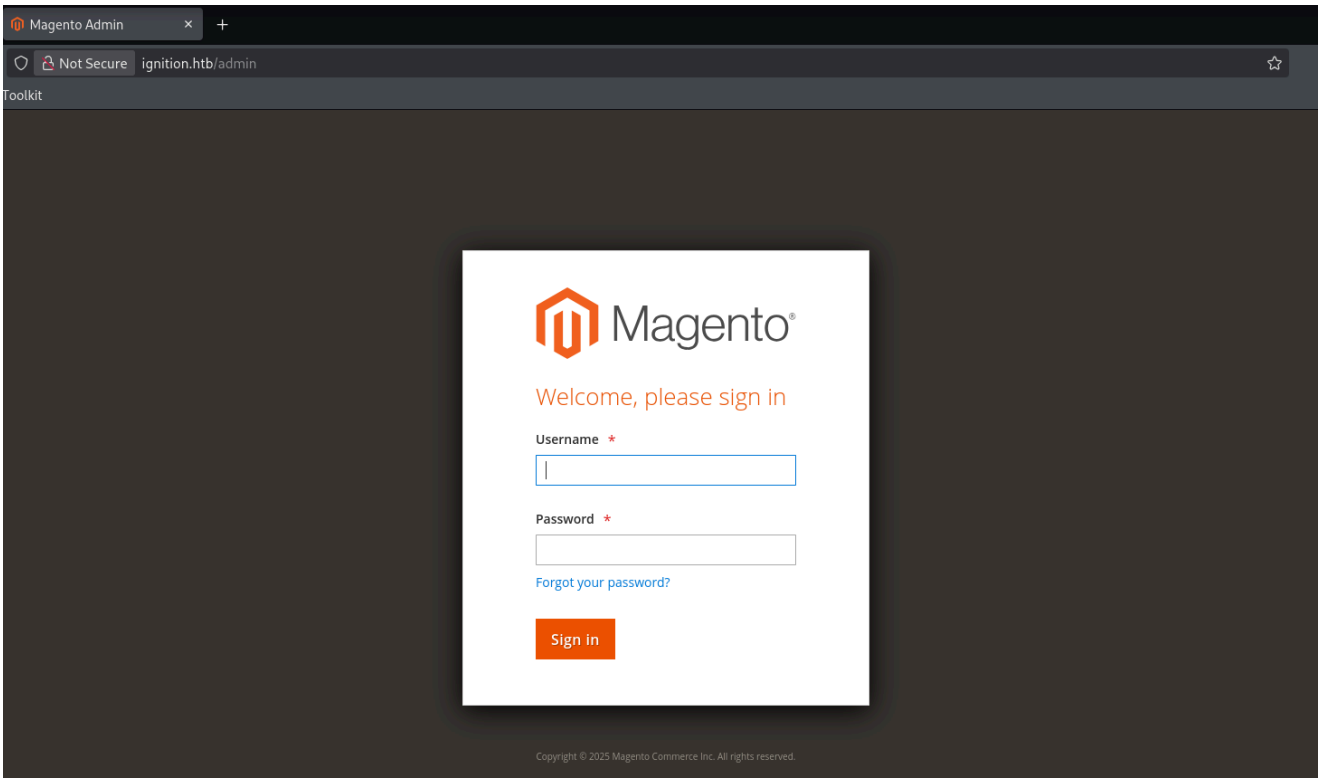


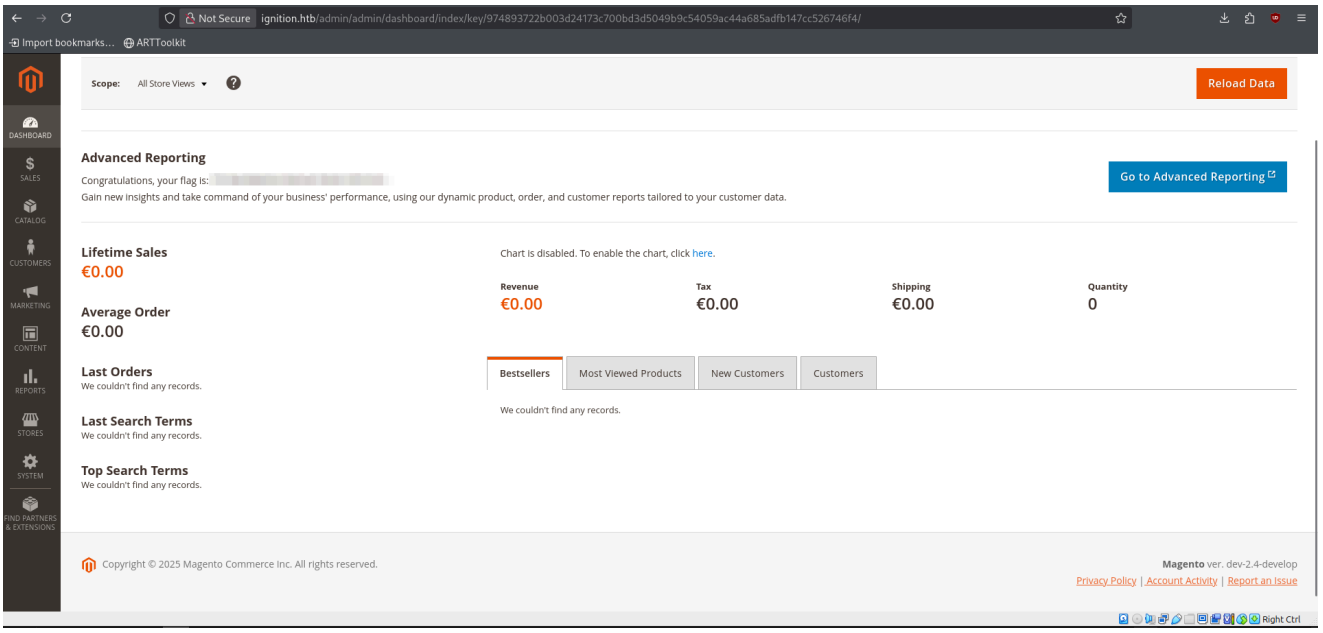


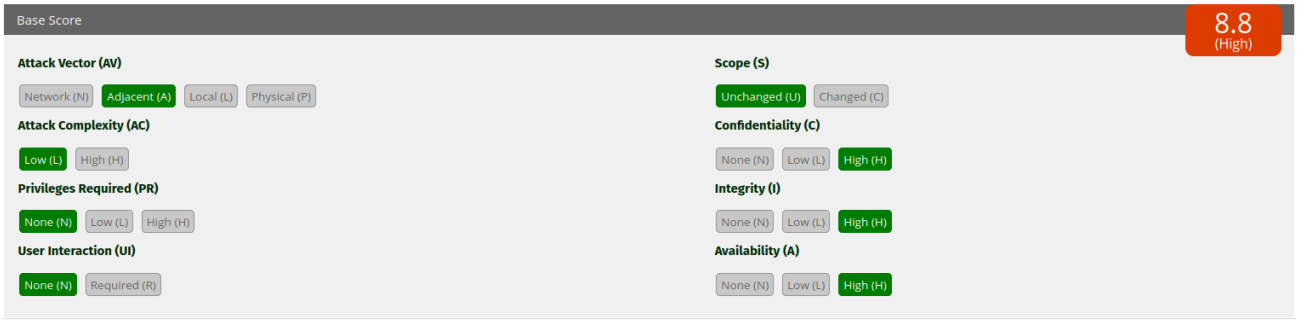
7. Evidence Collection

- **Objective:** Document and confirm all findings.
- **Procedure:** Screenshots were taken at critical junctures, including:
  - The initial ping confirming host availability.
  - Nmap outputs showing port and service details.
  - ffuf directory scan results revealing critical endpoints.
  - The successful login to the Magento admin panel.
- **Observation:** The collected evidence substantiates each step of the assessment, confirming the vulnerabilities and demonstrating the practical impact of weak security controls.

Vulnerability 1: Weak Authentication in Magento Admin Portal







- **CVSS v3.1 Base Score: 8.8 (High) Metric Breakdown:** AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
- **Description:** A significant vulnerability was identified in the Magento administration portal. The login interface relies on a weak password policy that requires only a minimum

of seven characters with a combination of letters and numbers. This leniency in credential requirements allows attackers to use common or predictable passwords to gain unauthorized access.

- **Impact:** Exploiting this vulnerability enables an attacker to infiltrate the Magento admin portal. Once authenticated, the attacker can modify site configurations, manage sensitive commerce data, and potentially escalate privileges throughout the application. This can lead to severe consequences, including the compromise of the confidentiality, integrity, and availability of critical business functions.
- **Technical Details:**
  - **Discovery:** Through directory enumeration with ffuf, the `/admin` endpoint was discovered. Subsequent analysis of the login page revealed that the password policy enforced only a basic requirement of at least seven alphanumeric characters.
  - **Access Exploitation:** By testing a list of common passwords, the password `qwerty123` was found to be valid, allowing unauthorized access:

```
Username: admin
Password: <REDACTED>
```

- **Evidence:** The provided screenshots capture the Magento admin portal interface and the successful login process, clearly demonstrating how easily the authentication mechanism can be bypassed using weak credentials.

4 Recommendations

To mitigate the vulnerability associated with weak authentication in the Magento administration portal, the following steps should be taken:

- **Enhance Password Policies:** Enforce a robust password policy for all administrative accounts. This should include requirements for a minimum password length of at least 12 characters, along with a mix of uppercase letters, lowercase letters, numbers, and special characters. Strengthening the password policy will reduce the likelihood of successfully guessing or brute-forcing credentials.
- **Implement Multi-Factor Authentication (MFA):** Introduce MFA for accessing the Magento admin portal. By requiring a second factor—such as a time-based OTP (One-Time Password) or hardware token—even if credentials are compromised, unauthorized access can be significantly curtailed.
- **Remove Default and Weak Credentials:** Immediately replace any default credentials and perform a routine audit to ensure no weak, common, or easily guessable passwords are in use. Regular updates to credentials, combined with the enforcement of strong password policies, will minimize the vulnerability to credential-based attacks.
- **Restrict Access to the Admin Interface:** Limit access to the Magento administration portal by implementing IP whitelisting or VPN-only access. Restricting access to trusted

networks will reduce exposure to unauthorized login attempts and further secure the administrative interface.
- **Continuous Monitoring and Regular Security Audits:** Establish continuous monitoring and logging to promptly detect anomalous access attempts to the admin portal. Complement this with regular security audits and vulnerability assessments to ensure that any emerging threats or authentication weaknesses are identified and mitigated in a timely fashion.

By taking these steps, the risk associated with weak authentication in the Magento admin portal will be significantly mitigated, thereby protecting sensitive business data and maintaining a strong overall security posture.

5 Conclusions

Executive Summary

Imagine a state-of-the-art facility where every door is securely locked—except for one seemingly insignificant entry that remains ajar. This vulnerable access point could allow an intruder to bypass your defenses and infiltrate the most sensitive areas without detection. In our assessment of your Magento-based web application hosted on ignition.htb, we discovered that the administrative interface is effectively “unlocked” due to weak authentication measures. An attacker could easily gain unauthorized access using common or predictable credentials, potentially leading to data manipulation, operational disruptions, financial losses, and long-term reputational damage. Immediate action is crucial to fortify this weak point and protect your valuable assets.

Technical Summary

Our investigation revealed a critical misconfiguration in the Magento admin portal's authentication mechanism. In simpler terms, the login system does not enforce robust security standards, making it vulnerable to attacks that rely on weak, predictable passwords. Although the technical specifics involve details such as password complexity requirements and configuration settings, the real-world impact is straightforward: unauthorized access to the admin panel can result in the compromise of sensitive data, unauthorized changes to site configurations, and a cascade of negative consequences for your business operations.

Current Security Posture and Future Steps

Risk Assessment: The current vulnerability in the Magento admin portal represents a significant risk that could jeopardize sensitive information, disrupt core business functions, and erode trust among customers and partners.

Recommended Actions:

1. **Immediate Remediation:**

- **Seal the Vulnerable Access Point:** Implement strict access controls on the Magento admin portal by enforcing multi-factor authentication (MFA) and a robust password policy that requires complex, unique credentials.
- **Remove Weak Credentials:** Undertake an immediate audit of existing administrative accounts to identify and replace any weak or default passwords.

2. **Enhanced Security Measures:**
   - **Layered Defenses:** Adopt additional security layers such as IP whitelisting, network segmentation, and periodic security audits to ensure that even if one control is bypassed, other defenses will remain active.
   - **Employee Security Awareness:** Enhance security training programs to underscore the importance of secure password practices and proper administrative procedures across the organization.
3. **Ongoing Monitoring:**
   - Establish continuous monitoring and logging to quickly identify and respond to suspicious activity. Regular vulnerability assessments and penetration tests should be conducted to stay ahead of emerging threats.

By addressing these vulnerabilities promptly and investing in a comprehensive security strategy, your organization can substantially reduce the risk of unauthorized access. This proactive approach not only protects your operational integrity and financial stability but also reinforces the trust that your customers and partners place in your organization in today's increasingly digital landscape.

Appendix: Tools Used

This section details the primary tools used during the assessment, along with a brief explanation of each. These tools provided crucial insights into the target system's configuration, identified exposed services, and enabled the demonstration of security weaknesses.

- **Ping Description:** Ping is a basic network diagnostic utility that checks host availability and measures network latency. In our assessment, Ping confirmed that the target system was online and helped to infer that it is Linux-based by analyzing the observed TTL value.
- **Nmap Description:** Nmap is a versatile network scanning tool used to discover active hosts, open ports, and running services. It was critical in mapping the target's network surface, revealing that the web service on port 80 was active and prompting further investigation.
- **WhatWeb Description:** WhatWeb is a web technology fingerprinting tool that identifies frameworks and server information. In our assessment, it helped determine that the underlying technology powering the site was Magento.
- **ffuf Description:** ffuf is a fast web fuzzer used for enumerating directories and endpoints. It played an essential role in discovering hidden directories such as the

Magento admin interface, which was later analyzed to expose weak authentication measures.