Analysis of HTTP 403 Bypass Tools

Eyad El-Taher

January 22, 2026

1 Tool 1: Bypass-403

GitHub: `https://github.com/iamj0ker/bypass-403`

1.1 Description

A lightweight shell script designed for security researchers to bypass 403 Forbidden restrictions. It allows users to compare server responses under various conditions to identify misconfigurations.

1.2 Installation

To set up the environment and dependencies:

```
git clone https://github.com/iamj0ker/bypass-403
cd bypass-403
chmod +x bypass-403.sh
sudo apt install figlet jq
```

1.3 Usage

```
./bypass-403.sh https://example.com admin
./bypass-403.sh [website] [path]
```

1.4 Features

- Utilizes 24 known bypass techniques.
- Leverages `curl` for request automation.
- **Contributors:** remonsec, manpreetMayankPandey01, saadibabar.

2 Tool 2: 4-ZERO-3

GitHub: `https://github.com/Dheerajmadhukar/4-ZERO-3`

2.1 Introduction

4-ZERO-3 is a comprehensive script designed to bypass 403/401 unauthorized errors. It incorporates a wide array of techniques and provides the exact `cURL` payload when a successful bypass is detected.

> **Note on False Positives:** If multiple [200 OK] responses are received, verify the `Content-Length`. Identical lengths often indicate a false positive caused by redirects (301/302).

2.2 Operational Modes

The tool supports specific flags for targeted testing:

- **Protocol:** `--protocol` (Protocol-based payloads)
- **Port:** `--port` (Port-based payloads)
- **HTTP Methods:** `--HTTPmethod` (Verb tampering)
- **Encoding:** `--encode` (URL encoded bypasses)
- **SQLi:** `--SQLi` (Mod_Security & libinjection bypasses)
- **Headers:** `--header` (Header-based manipulation)
- **Complete scan:** `--exploit`

2.3 Usage Examples

Execute a complete scan against an endpoint:

```
bash 403-bypass.sh -u https://target.com/secret --exploit
```

2.4 Prerequisites

- `apt install curl` (Debian-based systems)