

OWASP ZAP Scan Report – google gruyere

https://google-gruyere.appspot.com



April 16, 2024

Tyler Provencher

https://www.linkedin.com/in/tyler-provencher/

https://www.provenchermultimedia.com

# A blue and white logo Description automatically generated

Table of Contents

[0](#_Toc164264203)

[Section 1: Executive Summary 2](#_Toc164264204)

[1.1: Key Findings 2](#_Toc164264205)

[Section 2: Introduction 3](#_Toc164264206)

[2.1: Target Application: Google Gruyere 3](#_Toc164264207)

[Section 3: Scope of Scan 4](#_Toc164264208)

[Section 4: Methodology 5](#_Toc164264209)

[4.1: Scanning Environment 5](#_Toc164264210)

[4.2: Testing Approach 6](#_Toc164264211)

[4.3: Scan Depth 7](#_Toc164264212)

[4.4: Scan Policies 7](#_Toc164264213)

[Section 5: Scan Findings 9](#_Toc164264214)

[5.1: High Severity Vulnerabilities 10](#_Toc164264215)

[5.2: Medium Severity Vulnerabilities 11](#_Toc164264216)

[5.3: Low Severity Vulnerabilities 13](#_Toc164264217)

[Section 6: Vulnerability Details 13](#_Toc164264218)

[Section 7: Recommendations 13](#_Toc164264219)

[Section 8: Conclusion 13](#_Toc164264220)

[Section 9: Appendix 13](#_Toc164264221)

[9.1 Affected URLs: 13](#_Toc164264222)

# Section 1: **Executive Summary**

The OWASP ZAP scan conducted on the Google Gruyere web application revealed several key findings regarding security vulnerabilities. The scan primarily focused on identifying potential risks within the application's functionality and architecture.

## 1.1: Key Findings

The following vulnerabilities were identified during the OWASP Zap scan of the Google Gruyere web application, categorized by their corresponding risk severity levels:

**High Severity Vulnerabilities Found:**

* Cross Site Scripting (Reflected)

**Medium Severity Vulnerabilities Found:**

* Absence of Anti-CSRF Tokens
* Content Security Policy (CSP) Header Not Set
* Missing Anti-clickjacking Header

**Low Severity Vulnerabilities Found:**

* Cookie No HttpOnly Flag
* Cookie without SameSite Attribute
* Cross-Domain JavaScript Source File Inclusion
* Strict-Transport-Security Header Not Set
* X-Content-Type-Options Header Missing

**Informational Risk Severity Level Vulnerabilities Found:**

* Charset Mismatch (Header Versus Meta Content-Type Charset)
* Cookie Poisoning
* Information Disclosure - Suspicious Comments
* Modern Web Application
* Re-examine Cache-control Directives
* Retrieved from Cache
* Session Management Response Identified
* User Agent Fuzzer

The detailed analysis and explanations for the identified vulnerabilities will be provided in Sections 5 and 6 of this report. Additionally, actionable recommendations and mitigation strategies to address these vulnerabilities effectively will be outlined in Sections 5 and 7.

# Section 2: Introduction

The purpose of this scan is twofold: to serve as an academic research endeavor aimed at gaining hands-on experience with web application scanning, report writing, and cybersecurity best practices, and to contribute to a portfolio-building cybersecurity project. This report is my first attempt at conducting a comprehensive web application security scan and compiling a detailed vulnerability assessment report.

## 2.1: Target Application: Google Gruyere

Google Gruyere (https://google-gruyere.appspot.com) is a deliberately vulnerable web application designed by Google to illustrate common web security vulnerabilities and serve as an educational platform for cybersecurity enthusiasts. It features a range of vulnerabilities, including but not limited to SQL injection, cross-site scripting (XSS), and insecure configuration settings.

The objective of scanning Google Gruyere is to identify and analyze these vulnerabilities using OWASP ZAP, a widely recognized security testing tool, and to document the findings in a structured report format. This project will hopefully enhance my understanding of web application security principles.

As this is my first scan and report in a professional capacity, I approach this endeavor with a commitment to professionalism, accuracy, and thoroughness. The insights gained from this scan will strengthen my capabilities as a cybersecurity professional.

# Section 3: Scope of Scan

**The scope of the OWASP ZAP scan conducted on the Google Gruyere website (https://google-gruyere.appspot.com) encompassed the following:**

1. **URLs:**

All publicly accessible URLs within the Google Gruyere domain were included in the scan. This includes:

* <https://google-gruyere.appspot.com/>
* Any additional paths and pages within the domain structure.
* 223 total URLs scanned.

1. **Parameters:**

The scan targeted all parameters within the URLs, including query parameters, form inputs, and any other user-supplied inputs processed by the web application.

1. **Technologies Tested:**

The scan covered a range of technologies commonly used in web applications, including but not limited to:

* Front-end programming languages: HTML, CSS, JavaScript
* Server-side scripting languages (e.g., Python, Java)
* Web application frameworks and libraries

# Section 4: Methodology

## 4.1: Scanning Environment

The OWASP ZAP scan was conducted using a dedicated scanning environment configured as follows:

* **Operating System:** Kali Linux 2024.1 (Debian 64-bit)
* **Virtualization Platform:** Oracle VM VirtualBox
* **Virtual Machine Configuration:**
  + **OS:** Kali Linux 2024.1 (Debian 64-bit)
  + **CPU:** AMD64 architecture
  + **Memory:** 4096 MB
  + **Disk Space:** 25 GB

The use of Kali Linux provides a comprehensive suite of security tools and utilities, including OWASP ZAP, for conducting thorough web application security assessments. Oracle VM VirtualBox was selected as the virtualization platform to create an isolated and controlled scanning environment without impacting the host system.

**Additional Configuration Details:**

* **OWASP ZAP version:** 2.14.0
* **Proxy Settings:** OWASP ZAP configured as a proxy for intercepting and analyzing HTTP(S) traffic during the scan.

This scanning environment ensured a controlled and secure testing environment for conducting the OWASP ZAP scan of the Google Gruyere web application, allowing for accurate vulnerability identification and analysis.

## 4.2: Testing Approach

The testing approach for the OWASP ZAP scan conducted on the Google Gruyere web application involved utilizing automated scans for all vulnerability categories outlined in the comprehensive scan policy (section 4.3). This approach involved configuring OWASP ZAP to perform automated scans targeting a wide range of vulnerabilities, including but not limited to injection vulnerabilities, information disclosure, server security issues, and miscellaneous attack vectors.

The automated testing approach included the following steps for each vulnerability category:

* **Automated Scans:**
  + OWASP ZAP's active scanning capabilities were leveraged to automatically identify and test for vulnerabilities across multiple categories, including injections (SQL injection, XSS), information disclosure, server-side vulnerabilities, and miscellaneous attack vectors.
* **Policy-Based Scanning:**
  + The scan policy settings, as outlined previously in the methodology, were applied to ensure comprehensive coverage of potential security weaknesses and vulnerabilities within the Google Gruyere application.
* **Detection and Analysis:**
  + OWASP ZAP automatically detected and analyzed HTTP(S) requests and responses, looking for pattern’s indicative of known vulnerabilities such as SQL injection, XSS, path traversal, and more.
* **Reporting and Analysis:**
  + Upon completion of the automated scans, OWASP ZAP generated a detailed report highlighting the identified vulnerabilities, including severity levels, affected URLs, and potential impact.

This testing approach facilitated a comprehensive evaluation of the Google Gruyere web application's security posture, leveraging automated scans across diverse vulnerability categories to identify and prioritize potential security weaknesses and vulnerabilities.

## 4.3: Scan Depth

The OWASP ZAP scan conducted on the Google Gruyere web application was a surface-level scan focused on accessible URLs and parameters. This approach targeted the publicly available and commonly accessed areas of the application without delving into hidden or restricted areas.

**Key Focus Areas:**

* The scan prioritized scanning URLs and parameters accessible to regular users without requiring privileged access or specific authentication credentials.
* Commonly used functionalities and features of the Google Gruyere application were thoroughly scanned for potential security vulnerabilities, including injection attacks, information disclosure, and server-side vulnerabilities.

**Exclusion of Hidden or Restricted Areas:**

* The scan did not include deep scanning of hidden or restricted areas of the application that would typically require specialized permissions or authentication beyond standard user access.

By focusing on surface-level scanning, the objective was to identify and analyze security vulnerabilities and weaknesses in the publicly accessible components of the Google Gruyere web application, providing insights into potential risks that could affect regular users interacting with the application.

## 4.4: Scan Policies

The OWASP ZAP scan was conducted with a comprehensive scan policy designed to cover a wide range of security vulnerabilities and attack vectors. The scan policy settings used for this assessment are outlined below:

* **Client-Browser:**
  + Cross Site Scripting (DOM Based)
* **Information Gathering:**
  + .env Information Leak
  + .htaccess Information Leak
  + Directory Browsing
  + ELMAH Information Leak
  + Heartbleed OpenSSL Vulnerability
  + Hidden File Finder
  + Remote Code Execution - CVE-2012-1823
  + Source Code Disclosure - /WEB-INF Folder
  + Source Code Disclosure - CVE-2012-1823
  + Spring Actuator Information Leak
  + Trace.axd Information Leak
  + User Agent Fuzzer
* **Injection:**
  + Buffer Overflow
  + Cloud Metadata Potentially Exposed
  + CRLF Injection
  + Cross Site Scripting (Persistent)
  + Cross Site Scripting (Persistent) – Prime
  + Cross Site Scripting (Persistent) – Spider
  + Cross Site Scripting (Reflected)
  + Format String Error
  + Parameter Tampering
  + Remote OS Command Injection
  + Server Side Code Injection
  + Server Side Include
  + Server Side Template Injection
  + Server Side Template Injection (Blind)
  + Spring4Shell
  + SQL Injection
  + SQL Injection - Hypersonic SQL
  + SQL Injection – MsSQL
  + SQL Injection – MySQL
  + SQL Injection – Oracle
  + SQL Injection – PostgreSQL
  + SQL Injection – SQLite
  + XML External Entity Attack
  + XPath Injection
  + XSLT Injection
* **Miscellaneous:**
  + External Redirect
  + Generic Padding Oracle
  + GET for POST
  + Log4Shell
  + Script Active Scan Rules
  + SOAP Action Spoofing (Beta)
  + SOAP XML Injection (Beta)
* **Server Security:**
  + Path Traversal
  + Remote File Inclusion

Each vulnerability category was scanned using default thresholds and strengths, with a focus on identifying potential security weaknesses and vulnerabilities within the Google Gruyere web application.

# Section 5: Scan Findings

This section presents a detailed analysis of the vulnerabilities discovered during the OWASP ZAP scan of the Google Gruyere web application. The findings are categorized based on their severity levels, ranging from high-risk vulnerabilities that require immediate attention to lower-risk issues that still warrant mitigation measures. Each vulnerability is described with specific details, including affected URLs, parameters, potential impacts, and recommended remediation steps or mitigation strategies. The insights provided here aim to assist in understanding the security posture of the application and guide towards enhancing its resilience against potential threats.

## 5.1: High Severity Vulnerabilities

These vulnerabilities are categorized as high severity due to their potential to cause significant harm or compromise to the system. They require immediate attention and remediation to prevent exploitation by malicious actors.

1. **Cross Site Scripting (Reflected)**
   * **URL:** <http://google-gruyere.appspot.com/635819304277645928474860676940086211620/snippets.gtl?uid=%3C%2Fh2%3E%3CscrIpt%3Ealert%281%29%3B%3C%2FscRipt%3E%3Ch2%3E>
   * **Risk:** High
   * **Confidence:** Medium
   * **Parameter:** uid
   * **Description:** Cross-Site Scripting (XSS) is a high-risk vulnerability that allows attackers to inject malicious scripts into web pages viewed by other users. In this case, the absence of input sanitization in the URL parameter 'uid' of the snippets page exposes users to potential script execution attacks. This vulnerability could allow an attacker to execute arbitrary code in a user's browser instance.
   * **Recommendation:** Immediate remediation steps include implementing proper input validation and output encoding mechanisms to mitigate XSS attacks. Refer to OWASP XSS Prevention Cheat Sheet for detailed mitigation strategies.
   * **A screenshot of a computer

     Description automatically generatedReference:** <https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html>

## 5.2: Medium Severity Vulnerabilities

These vulnerabilities are considered medium severity as they pose a moderate risk to the system's security and functionality. While not as critical as high severity vulnerabilities, they still need to be addressed promptly to minimize potential threats.

1. **Absence of Anti-CSRF Tokens**

* **URL:** <http://google-gruyere.appspot.com/635819304277645928474860676940086211620/newaccount.gtl> & 6 other URLs.
* **Risk:** Medium
* **Confidence:** Low
* **Description:** The absence of Anti-CSRF Tokens in HTML forms leaves the application vulnerable to Cross-Site Request Forgery (CSRF) attacks. Without proper token validation, attackers could forge requests on behalf of authenticated users, leading to unauthorized actions**.** No Anti-CSRF tokens found in a HTML submission form, exposing the application to Cross-Site Request Forgery (CSRF) attacks.
* **Recommendation:** Immediate mitigation steps include implementing Anti-CSRF tokens and ensuring secure session management practices. Refer to OWASP CSRF Prevention Cheat Sheet for detailed mitigation strategies.
* **Reference:** <https://cheatsheetseries.owasp.org/cheatsheets/Cross-Site_Request_Forgery_Prevention_Cheat_Sheet.html>

A screenshot of a computer code

Description automatically generated

1. **Content Security Policy (CSP) Header Not Set**
   * **URL:** <https://google-gruyere.appspot.com/7> & 113 other URLs.
   * **Risk:** Medium
   * **Confidence:** High
   * **Description:** Missing Content Security Policy (CSP) header, leaving the application vulnerable to various attacks such as Cross-Site Scripting (XSS).
   * **Recommendation:** Implementing CSP headers can help mitigate such risks significantly. Configure the web server to set the Content-Security-Policy header. Refer to OWASP CSP Cheat Sheet for guidance.
   * **Reference:** <https://cheatsheetseries.owasp.org/cheatsheets/Content_Security_Policy_Cheat_Sheet.html>

**A screenshot of a computer program

Description automatically generated**

1. **Missing Anti-clickjacking Header**
   * **URL:** <http://google-gruyere.appspot.com/2> & 111 other URLs.
   * **Risk:** Medium
   * **Confidence:** Medium
   * **Description:** Absence of Anti-clickjacking headers exposes the application to ClickJacking attacks, where malicious entities can trick users into interacting with hidden or disguised elements.
   * **Recommendation:** Set X-Frame-Options or Content-Security-Policy headers to protect against ClickJacking. Refer to OWASP Clickjacking Prevention Cheat Sheet (9.1.4) for mitigation strategies.

**Reference:** <https://owasp.org/www-project-web-security-testing-guide/v42/4-Web_Application_Security_Testing/11-Client-side_Testing/09-Testing_for_Clickjacking>

A screenshot of a computer program

Description automatically generated

## 5.3: Low Severity Vulnerabilities

Low severity vulnerabilities are typically less urgent but still require attention and mitigation measures. Although they may not pose an immediate threat, addressing them helps improve overall system security and resilience against potential attacks.

1. **Cookie No HttpOnly Flag**
   * **URL:** <http://google-gruyere.appspot.com/start>
   * **Risk:** Low
   * **Confidence:** Medium
   * **Parameter:** GRUYERE\_ID
   * **Description:** A cookie has been set without the HttpOnly flag, which means that the cookie can be accessed by JavaScript. If a malicious script can be run on this page, then the cookie will be accessible and can be transmitted to another site. If this is a session cookie, then session hijacking may be possible.
   * **Solution:** Ensure that the HttpOnly flag is set for all cookies to prevent client-side script access.
   * **Reference:** <https://owasp.org/www-community/HttpOnly>

**A white background with black text

Description automatically generated**

1. **Cookie without SameSite Attribute**
   * **URL:** <http://google-gruyere.appspot.com/start>
   * **Risk:** Low
   * **Confidence:** Medium
   * **Parameter:** GRUYERE\_ID
   * **Description:** A cookie has been set without the SameSite attribute, which means that the cookie can be sent as a result of a 'cross-site' request. The SameSite attribute is an effective countermeasure to cross-site request forgery, cross-site script inclusion, and timing attacks.
   * **Solution:** Ensure that the SameSite attribute is set to either 'lax' or ideally 'strict' for all cookies to prevent cross-site request vulnerabilities.
   * **Reference:** <https://tools.ietf.org/html/draft-ietf-httpbis-cookie-same-site>

**A white background with black text

Description automatically generated**

1. **Cross-Domain JavaScript Source File Inclusion**
   * **URL:** <https://google-gruyere.appspot.com/resetbutton/635819304277645928474860676940086211620>
   * **Risk:** Low
   * **Confidence:** Medium
   * **Parameter:** <https://www.google.com/recaptcha/api.js>
   * **Description:** The page includes one or more script files from a third-party domain, potentially introducing security risks such as script injection and data leakage.
   * **Solution:** Ensure JavaScript source files are loaded from only trusted sources to mitigate the risk of cross-domain script inclusion vulnerabilities.
   * **Reference:** OWASP\_2021\_A08 - <https://owasp.org/Top10/A08_2021-Software_and_Data_Integrity_Failures/>

A screenshot of a computer code

Description automatically generated

1. **Strict-Transport-Security Header Not Set**
   * **URL:** <https://google-gruyere.appspot.com/start>.
   * **Risk:** Low
   * **Confidence:** High
   * **Description:** HTTP Strict Transport Security (HSTS) header is not set, leaving the application vulnerable to downgrade attacks and protocol downgrade risks.
   * **Solution:** Configure the web server, application server, load balancer, etc., to enforce Strict-Transport-Security to ensure all communications are conducted over secure HTTPS connections.
   * **Reference:** OWASP\_2021\_A05 - <https://owasp.org/Top10/A05_2021-Security_Misconfiguration/>

A white background with black text

Description automatically generated

1. **X-Content-Type-Options Header Missing**
   * **URL:** <http://google-gruyere.appspot.com/static/codelab.css>
   * **Risk:** Low
   * **Confidence:** Medium
   * **Description:** The X-Content-Type-Options header is missing, allowing older versions of Internet Explorer and Chrome to perform MIME-sniffing on the response body. This could lead to misinterpretation and display of the response body as a content type other than the declared one, potentially exposing security vulnerabilities.
   * **Solution:** Ensure that the application/web server sets the Content-Type header appropriately and includes the X-Content-Type-Options header set to 'nosniff' for all web pages.
   * **Reference:** OWASP\_2021\_A05 - Security Misconfiguration

A computer code with black text

Description automatically generated

## 5.4: Informational Severity Vulnerabilities

Informational severity level vulnerabilities, while not posing an immediate threat, provide valuable insights and recommendations to enhance system security and reduce potential risks.

# Section 6: Vulnerability Details

# Section 7: Recommendations

# Section 8: Conclusion

# Section 9: Appendix

## 9.1 Affected URLs:

This section includes all the affected URLs of the web application separated by the vulnerability.

1. **OWASP XSS Prevention Cheat Sheet:**
   * GET: <http://google-gruyere.appspot.com/635819304277645928474860676940086211620/snippets.gtl?uid=%3C%2Fh2%3E%3CscrIpt%3Ealert%281%29%3B%3C%2FscRipt%3E%3Ch2%3E>
2. **Absence of Anti-CSRF Tokens:**

* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/login
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/login?pw=ZAP&uid=ZAP
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/newaccount.gtl
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/login
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/newaccount.gtl
* GET: https://google-gruyere.appspot.com/resetbutton/635819304277645928474860676940086211620

1. **Content Security Policy (CSP) Header Not Set:**

* GET: http://google-gruyere.appspot.com
* GET: http://google-gruyere.appspot.com/
* GET: http://google-gruyere.appspot.com/0
* GET: http://google-gruyere.appspot.com/1
* GET: http://google-gruyere.appspot.com/2
* GET: http://google-gruyere.appspot.com/3
* GET: http://google-gruyere.appspot.com/4
* GET: http://google-gruyere.appspot.com/5
* GET: http://google-gruyere.appspot.com/6
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/login
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/login?pw=ZAP&uid=ZAP
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/newaccount.gtl
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/saveprofile?action=new&is\_author=True&pw=ZAP&uid=ZAP
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/snippets.gtl?uid=brie
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/snippets.gtl?uid=cheddar
* GET: http://google-gruyere.appspot.com/7
* GET: http://google-gruyere.appspot.com/8
* GET: http://google-gruyere.appspot.com/9
* GET: http://google-gruyere.appspot.com/code/
* GET: http://google-gruyere.appspot.com/code/?data.py
* GET: http://google-gruyere.appspot.com/code/?gruyere.py
* GET: http://google-gruyere.appspot.com/code/?gtl.py
* GET: http://google-gruyere.appspot.com/code/?resoources/dump.gtl
* GET: http://google-gruyere.appspot.com/code/?resources/dump.gtl
* GET: http://google-gruyere.appspot.com/code/?resources/editprofile.gtl
* GET: http://google-gruyere.appspot.com/code/?resources/error.gtl
* GET: http://google-gruyere.appspot.com/code/?resources/feed.gtl
* GET: http://google-gruyere.appspot.com/code/?resources/home.gtl
* GET: http://google-gruyere.appspot.com/code/?resources/manage.gtl
* GET: http://google-gruyere.appspot.com/code/?resources/menubar.gtl
* GET: http://google-gruyere.appspot.com/code/?sanitize.py
* GET: http://google-gruyere.appspot.com/code/data.py
* GET: http://google-gruyere.appspot.com/code/gruyere.py
* GET: http://google-gruyere.appspot.com/code/gtl.py
* GET: http://google-gruyere.appspot.com/code/resources/base.css
* GET: http://google-gruyere.appspot.com/code/resources/dump.gtl
* GET: http://google-gruyere.appspot.com/code/resources/editprofile.gtl
* GET: http://google-gruyere.appspot.com/code/resources/error.gtl
* GET: http://google-gruyere.appspot.com/code/resources/feed.gtl
* GET: http://google-gruyere.appspot.com/code/resources/home.gtl
* GET: http://google-gruyere.appspot.com/code/resources/lib.js
* GET: http://google-gruyere.appspot.com/code/resources/login.gtl
* GET: http://google-gruyere.appspot.com/code/resources/manage.gtl
* GET: http://google-gruyere.appspot.com/code/resources/menubar.gtl
* GET: http://google-gruyere.appspot.com/code/resources/newaccount.gtl
* GET: http://google-gruyere.appspot.com/code/resources/newsnippet.gtl
* GET: http://google-gruyere.appspot.com/code/resources/showprofile.gtl
* GET: http://google-gruyere.appspot.com/code/resources/snippets.gtl
* GET: http://google-gruyere.appspot.com/code/resources/upload.gtl
* GET: http://google-gruyere.appspot.com/code/resources/upload2.gtl
* GET: http://google-gruyere.appspot.com/code/sanitize.py
* GET: http://google-gruyere.appspot.com/code/secret.txt
* GET: http://google-gruyere.appspot.com/part1
* GET: http://google-gruyere.appspot.com/part2
* GET: http://google-gruyere.appspot.com/part3
* GET: http://google-gruyere.appspot.com/part4
* GET: http://google-gruyere.appspot.com/part5
* GET: http://google-gruyere.appspot.com/start
* GET: http://google-gruyere.appspot.com/static/codeindex.html
* GET: http://google-gruyere.appspot.com/static/codeindex/html
* GET: https://google-gruyere.appspot.com/
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/feed.gtl
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/login
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/newaccount.gtl
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/quitserver.
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/RESET.
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/saveprofile?action=update&is\_admin=True
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/saveprofile?action=update&is\_admin=True&uid=username
* GET: https://google-gruyere.appspot.com/code/
* GET: https://google-gruyere.appspot.com/code/?data.py
* GET: https://google-gruyere.appspot.com/code/?gruyere.py
* GET: https://google-gruyere.appspot.com/code/?gtl.py
* GET: https://google-gruyere.appspot.com/code/?resoources/dump.gtl
* GET: https://google-gruyere.appspot.com/code/?resources/dump.gtl
* GET: https://google-gruyere.appspot.com/code/?resources/editprofile.gtl
* GET: https://google-gruyere.appspot.com/code/?resources/error.gtl
* GET: https://google-gruyere.appspot.com/code/?resources/feed.gtl
* GET: https://google-gruyere.appspot.com/code/?resources/home.gtl
* GET: https://google-gruyere.appspot.com/code/?resources/manage.gtl
* GET: https://google-gruyere.appspot.com/code/?resources/menubar.gtl
* GET: https://google-gruyere.appspot.com/code/?sanitize.py
* GET: https://google-gruyere.appspot.com/code/data.py
* GET: https://google-gruyere.appspot.com/code/gruyere.py
* GET: https://google-gruyere.appspot.com/code/gtl.py
* GET: https://google-gruyere.appspot.com/code/resources/base.css
* GET: https://google-gruyere.appspot.com/code/resources/dump.gtl
* GET: https://google-gruyere.appspot.com/code/resources/editprofile.gtl
* GET: https://google-gruyere.appspot.com/code/resources/error.gtl
* GET: https://google-gruyere.appspot.com/code/resources/feed.gtl
* GET: https://google-gruyere.appspot.com/code/resources/home.gtl
* GET: https://google-gruyere.appspot.com/code/resources/lib.js
* GET: https://google-gruyere.appspot.com/code/resources/login.gtl
* GET: https://google-gruyere.appspot.com/code/resources/manage.gtl
* GET: https://google-gruyere.appspot.com/code/resources/menubar.gtl
* GET: https://google-gruyere.appspot.com/code/resources/newaccount.gtl
* GET: https://google-gruyere.appspot.com/code/resources/newsnippet.gtl
* GET: https://google-gruyere.appspot.com/code/resources/showprofile.gtl
* GET: https://google-gruyere.appspot.com/code/resources/snippets.gtl
* GET: https://google-gruyere.appspot.com/code/resources/upload.gtl
* GET: https://google-gruyere.appspot.com/code/resources/upload2.gtl
* GET: https://google-gruyere.appspot.com/code/sanitize.py
* GET: https://google-gruyere.appspot.com/code/secret.txt
* GET: https://google-gruyere.appspot.com/part1
* GET: https://google-gruyere.appspot.com/part2
* GET: https://google-gruyere.appspot.com/part3
* GET: https://google-gruyere.appspot.com/part4
* GET: https://google-gruyere.appspot.com/part5
* GET: https://google-gruyere.appspot.com/resetbutton/635819304277645928474860676940086211620
* GET: https://google-gruyere.appspot.com/start
* GET: https://google-gruyere.appspot.com/static/codeindex.html
* GET: https://google-gruyere.appspot.com/static/codeindex/html
* POST: <https://google-gruyere.appspot.com/resetbutton/635819304277645928474860676940086211620>

1. **Missing Anti-clickjacking Header:**

* GET: http://google-gruyere.appspot.com
* GET: http://google-gruyere.appspot.com/
* GET: http://google-gruyere.appspot.com/0
* GET: http://google-gruyere.appspot.com/1
* GET: http://google-gruyere.appspot.com/2
* GET: http://google-gruyere.appspot.com/3
* GET: http://google-gruyere.appspot.com/4
* GET: http://google-gruyere.appspot.com/5
* GET: http://google-gruyere.appspot.com/6
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/login
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/login?pw=ZAP&uid=ZAP
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/newaccount.gtl
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/saveprofile?action=new&is\_author=True&pw=ZAP&uid=ZAP
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/snippets.gtl?uid=brie
* GET: http://google-gruyere.appspot.com/635819304277645928474860676940086211620/snippets.gtl?uid=cheddar
* GET: http://google-gruyere.appspot.com/7
* GET: http://google-gruyere.appspot.com/8
* GET: http://google-gruyere.appspot.com/9
* GET: http://google-gruyere.appspot.com/code/
* GET: http://google-gruyere.appspot.com/code/?data.py
* GET: http://google-gruyere.appspot.com/code/?gruyere.py
* GET: http://google-gruyere.appspot.com/code/?gtl.py
* GET: http://google-gruyere.appspot.com/code/?resoources/dump.gtl
* GET: http://google-gruyere.appspot.com/code/?resources/dump.gtl
* GET: http://google-gruyere.appspot.com/code/?resources/editprofile.gtl
* GET: http://google-gruyere.appspot.com/code/?resources/error.gtl
* GET: http://google-gruyere.appspot.com/code/?resources/feed.gtl
* GET: http://google-gruyere.appspot.com/code/?resources/home.gtl
* GET: http://google-gruyere.appspot.com/code/?resources/manage.gtl
* GET: http://google-gruyere.appspot.com/code/?resources/menubar.gtl
* GET: http://google-gruyere.appspot.com/code/?sanitize.py
* GET: http://google-gruyere.appspot.com/code/data.py
* GET: http://google-gruyere.appspot.com/code/gruyere.py
* GET: http://google-gruyere.appspot.com/code/gtl.py
* GET: http://google-gruyere.appspot.com/code/resources/base.css
* GET: http://google-gruyere.appspot.com/code/resources/dump.gtl
* GET: http://google-gruyere.appspot.com/code/resources/editprofile.gtl
* GET: http://google-gruyere.appspot.com/code/resources/error.gtl
* GET: http://google-gruyere.appspot.com/code/resources/feed.gtl
* GET: http://google-gruyere.appspot.com/code/resources/home.gtl
* GET: http://google-gruyere.appspot.com/code/resources/lib.js
* GET: http://google-gruyere.appspot.com/code/resources/login.gtl
* GET: http://google-gruyere.appspot.com/code/resources/manage.gtl
* GET: http://google-gruyere.appspot.com/code/resources/menubar.gtl
* GET: http://google-gruyere.appspot.com/code/resources/newaccount.gtl
* GET: http://google-gruyere.appspot.com/code/resources/newsnippet.gtl
* GET: http://google-gruyere.appspot.com/code/resources/showprofile.gtl
* GET: http://google-gruyere.appspot.com/code/resources/snippets.gtl
* GET: http://google-gruyere.appspot.com/code/resources/upload.gtl
* GET: http://google-gruyere.appspot.com/code/resources/upload2.gtl
* GET: http://google-gruyere.appspot.com/code/sanitize.py
* GET: http://google-gruyere.appspot.com/code/secret.txt
* GET: http://google-gruyere.appspot.com/part1
* GET: http://google-gruyere.appspot.com/part2
* GET: http://google-gruyere.appspot.com/part3
* GET: http://google-gruyere.appspot.com/part4
* GET: http://google-gruyere.appspot.com/part5
* GET: http://google-gruyere.appspot.com/start
* GET: http://google-gruyere.appspot.com/static/codeindex.html
* GET: https://google-gruyere.appspot.com/
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/feed.gtl
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/login
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/newaccount.gtl
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/quitserver.
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/RESET.
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/saveprofile?action=update&is\_admin=True
* GET: https://google-gruyere.appspot.com/635819304277645928474860676940086211620/saveprofile?action=update&is\_admin=True&uid=username
* GET: https://google-gruyere.appspot.com/code/
* GET: https://google-gruyere.appspot.com/code/?data.py
* GET: https://google-gruyere.appspot.com/code/?gruyere.py
* GET: https://google-gruyere.appspot.com/code/?gtl.py
* GET: https://google-gruyere.appspot.com/code/?resoources/dump.gtl
* GET: https://google-gruyere.appspot.com/code/?resources/dump.gtl
* GET: https://google-gruyere.appspot.com/code/?resources/editprofile.gtl
* GET: https://google-gruyere.appspot.com/code/?resources/error.gtl
* GET: https://google-gruyere.appspot.com/code/?resources/feed.gtl
* GET: https://google-gruyere.appspot.com/code/?resources/home.gtl
* GET: https://google-gruyere.appspot.com/code/?resources/manage.gtl
* GET: https://google-gruyere.appspot.com/code/?resources/menubar.gtl
* GET: https://google-gruyere.appspot.com/code/?sanitize.py
* GET: https://google-gruyere.appspot.com/code/data.py
* GET: https://google-gruyere.appspot.com/code/gruyere.py
* GET: https://google-gruyere.appspot.com/code/gtl.py
* GET: https://google-gruyere.appspot.com/code/resources/base.css
* GET: https://google-gruyere.appspot.com/code/resources/dump.gtl
* GET: https://google-gruyere.appspot.com/code/resources/editprofile.gtl
* GET: https://google-gruyere.appspot.com/code/resources/error.gtl
* GET: https://google-gruyere.appspot.com/code/resources/feed.gtl
* GET: https://google-gruyere.appspot.com/code/resources/home.gtl
* GET: https://google-gruyere.appspot.com/code/resources/lib.js
* GET: https://google-gruyere.appspot.com/code/resources/login.gtl
* GET: https://google-gruyere.appspot.com/code/resources/manage.gtl
* GET: https://google-gruyere.appspot.com/code/resources/menubar.gtl
* GET: https://google-gruyere.appspot.com/code/resources/newaccount.gtl
* GET: https://google-gruyere.appspot.com/code/resources/newsnippet.gtl
* GET: https://google-gruyere.appspot.com/code/resources/showprofile.gtl
* GET: https://google-gruyere.appspot.com/code/resources/snippets.gtl
* GET: https://google-gruyere.appspot.com/code/resources/upload.gtl
* GET: https://google-gruyere.appspot.com/code/resources/upload2.gtl
* GET: https://google-gruyere.appspot.com/code/sanitize.py
* GET: https://google-gruyere.appspot.com/code/secret.txt
* GET: https://google-gruyere.appspot.com/part1
* GET: https://google-gruyere.appspot.com/part2
* GET: https://google-gruyere.appspot.com/part3
* GET: https://google-gruyere.appspot.com/part4
* GET: https://google-gruyere.appspot.com/part5
* GET: https://google-gruyere.appspot.com/resetbutton/635819304277645928474860676940086211620
* GET: https://google-gruyere.appspot.com/start
* GET: https://google-gruyere.appspot.com/static/codeindex.html
* POST: https://google-gruyere.appspot.com/resetbutton/635819304277645928474860676940086211620