

## XSS Vulnerability in Uploader (plugin for WordPress) version 1.0.4

**Title:** XSS Vulnerability in Uploader (plugin for WordPress) version 1.0.4 **Dognædis Ref.:** DGS-SEC-16  
**Release Date:** 2013/03/01 **CVE Ref.:** CVE-2013-2287  
**Vulnerable Application:** Uploader plugin for WordPress (1.0.4) **Other Ref.:**  
**Type:** Cross Site Scripting **Discover Credits:** CodeV - Code Analyzer  
**Level:** High (Low/High/Critical) **Bulletin Author(s):** RMBR - CodeV Team  
**CVSS:** 4.9 (Av:N/AC:L/Au:S/C:C/I:P/A:N) **Contact:** irt@dognædis.com

### Nível de Acesso do Documento: Público

A informação expressa neste documento é propriedade da Dognædis. Pode ser, no entanto, lida, copiada, distribuída, impressa ou acedida por qualquer pessoa ou entidade, desde que os créditos da Dognædis sejam respeitados.

### Document Access Level: Public

The information expressed in this document is property of Dognædis. Although can be disclosed, distributed, copied, read, used, printed or accessed by anyone, since all the Dognædis credits are respected.

### Overview:

Uploader creates an Uploader role for file uploading.

### Scope:

**File:** <app\_root>/wp-content/plugins/uploader/views/notify.php

**Vulnerable Argument(s):** \$\_GET['notify']

\$\_GET['blog']

### Code:

```
line 26: echo $output;
```

### Proof(s) of Concept:

```
<app_root>/wp-content/plugins/uploader/views/notify.php?notify=unnotif&blog=%3Cscript%3Ealert%28123%29;%3C/script%3E
```

### Description:

Cross-site scripting (XSS) is a type of computer security vulnerability typically found in web applications that enables attackers to inject client-side script into web pages viewed by other users. A cross-site scripting vulnerability may be used by attackers to bypass access controls such as the same origin policy.

### Impact:

Generally, by exploiting this kind of vulnerability, it might be possible to achieve possible attack vectors to various kinds of attacks such as:

- Session/Cookie theft
- Account Hijacking
- Identity theft
- Accessing confidential resources
- Accessing pay content
- Account Denial of service

### Resolution:

Aiming a correct resolution of the identified vulnerability, the data obtained through the \$output input argument should be properly sanitized for HTML and following ECMAS usage.

### Official Solution:

At the moment, there is no official solution for the reported vulnerabilities.

**External References:**

[http://www.owasp.org/index.php/Cross-site\\_Scripting\\_\(XSS\)](http://www.owasp.org/index.php/Cross-site_Scripting_(XSS))

[http://www.owasp.org/index.php/Cross-Site\\_Request\\_Forgery\\_\(CSRF\)](http://www.owasp.org/index.php/Cross-Site_Request_Forgery_(CSRF))

[http://en.wikipedia.org/wiki/Cross-site\\_scripting](http://en.wikipedia.org/wiki/Cross-site_scripting)

[http://en.wikipedia.org/wiki/Code\\_injection](http://en.wikipedia.org/wiki/Code_injection)



DOGNÆDIS

---

TRUSTABLE SOLUTIONS