

XSS Vulnerability in Tomato Cart v1.1.8.1

Title: XSS Vulnerability in Tomato Cart v1.1.8.1

Release Date: 2013/03/01

Vulnerable Application: Tomato Cart v1.1.8.1

Type: Cross Site Scripting

Level: High (Low/High/Critical)

CVSS: 3.2 (Av:N/AC:M/Au:S/C:C/I:P/A:N)

Dognædis Ref.: DGS-SEC-11

CVE Ref.: CVE-2013-2282

Other Ref.:

Discover Credits: CodeV - Code Analyzer

Bulletin Author(s): LLaranja - CodeV Team

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:

TomatoCart is a next generation open source shopping cart, branched from osCommerce 3 as a separate project.

Scope:

File: app_root/templates/glass_gray/content/info/faqs.php

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

Code:

```
line 67: if(question.getParent().id == 'faq<?php echo $_GET['faqs_id']; ?>')
```

Proof(s) of Concept:

```
app_root/info.php?faqs&faqs_id=0'){});});alert(123);question.addEvent('click',  
function(e){question.addEvent('click', function(e){//
```

Description:

The referred vulnerabilities could be exploited through XSS (Cross-Site-Scripting) attacks.

Ultimately, the attacker could take complete control of the victims web-browser.

In a successful attack, the malicious script would be executed with the authenticated user permissions.

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 \$_GET['faqs_id'] 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/Code_injection



DOGNÆDIS

TRUSTABLE SOLUTIONS