

*The following security alert was issued by the Information Security Division of the Mississippi Department of ITS and is intended for State government entities. The information may or may not be applicable to the general public and accordingly, the State does not warrant its use for any specific purposes.*

**DATE(S) ISSUED:**

11/21/2014

**SUBJECT:**

Multiple Vulnerabilities in WordPress Content Management System

**EXECUTIVE SUMMARY:**

Multiple vulnerabilities have been discovered in WordPress CMS, which could allow an attacker to take control of the affected system. WordPress is an open source content management system (CMS) for websites.

Successful exploitation of the vulnerabilities could result in an attacker gaining un-authorized access, bypassing security restrictions, injecting scripts or HTML, and stealing cookies. Depending on the privileges gained, an attacker could install programs; view, change, or delete data; or create new accounts with full user rights.

**THREAT INTELLIGENCE:**

At this time, there is no known proof-of-concept code available.

**SYSTEM AFFECTED:**

- WordPress versions prior to 4.0.1

**RISK:**

**Government:**

- Large and medium government entities: **High**
- Small government entities: **High**

**Businesses:**

- Large and medium business entities: **High**
- Small business entities: **High**

**Home users: High**

#### **TECHNICAL SUMMARY:**

Three vulnerabilities have been identified in WordPress CMS that could allow for an attacker to take control of the affected system. Details of these vulnerabilities are as follows:

- A security-bypass vulnerability because it fails to properly validate the links in a password reset email. This may allow users to logs in, and changes their email address. Attackers can exploit this issue to bypass certain security restrictions to perform unauthorized actions.
- Multiple unspecified cross-site scripting vulnerabilities because it fails to properly sanitize user-supplied input. An attacker may leverage these issues to execute arbitrary script code in the browser of an unsuspecting user in the context of the affected site. This can allow the attacker to steal cookie-based authentication credentials and launch other attacks.
- An HTML-injection vulnerability because it fails to sufficiently sanitize user-supplied input submitted to the 'comment' field. Successful exploits will allow attacker-supplied HTML and script code to run in the context of the affected browser, potentially allowing the attacker to steal cookie-based authentication credentials or control how the site is rendered to the user. Other attacks are also possible.

Successful exploitation of these vulnerabilities could allow the attacker to bypass certain security restrictions, gain unauthorized access, run malicious HTML and script codes, or steal cookie-based authentication credentials. An attacker could then install programs; view, change, or delete data; or create new accounts with full user rights.

WordPress has released WordPress 4.0.1, which corrects these issues.

#### **RECOMMENDATIONS:**

The following actions should be taken:

- Update vulnerable systems running WordPress immediately after appropriate testing.
- Review and follow WordPress hardening guidelines - [http://codex.wordpress.org/Hardening\\_WordPress](http://codex.wordpress.org/Hardening_WordPress)
- Confirm that the operating system and all other applications on the system running this CMS are updated with the most recent patches.
- Deploy NIDS to detect and block attacks and anomalous activity such as crafted requests containing suspicious URI sequences.
- Run all software as a non-privileged user (one without administrative privileges) to diminish the effects of a successful attack.

#### **REFERENCES:**

**WordPress:**

<https://wordpress.org/news/2014/11/wordpress-4-0-1/>

**Security Focus:**

<http://www.securityfocus.com/bid/71231>

<http://www.securityfocus.com/bid/71236>

<http://www.securityfocus.com/bid/71237>