

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.

TLP: WHITE

Traffic Light Protocol (TLP): WHITE information may be distributed without restriction, subject to copyright controls.

<http://www.us-cert.gov/tlp/>

DATE(S) ISSUED:

09/19/2016

SUBJECT:

Multiple Vulnerabilities in PHP Could Allow for Arbitrary Code Execution

OVERVIEW:

Multiple vulnerabilities have been discovered in PHP, the most severe of which could allow an attacker to execute arbitrary code, with failed exploit attempts potentially leading to denial of service conditions. PHP is a programming language originally designed for use in web-based applications with HTML content. PHP supports a wide variety of platforms and is used by numerous web-based software applications. Successfully exploiting these issues could allow remote attackers to execute arbitrary code in the context of the affected application. Failed exploitation could result in a denial-of-service condition.

THREAT INTELLIGENCE:

There are currently no reports of these vulnerabilities being exploited in the wild. There is known proof-of-concept code for these vulnerabilities.

SYSTEM AFFECTED:

- PHP 5.6 prior to 5.6.26
- PHP 7 prior to 7.0.11

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: Low

TECHNICAL SUMMARY:

PHP has released updates that address multiple vulnerabilities, the most severe of which could allow for arbitrary code execution. These vulnerabilities include:

Prior to 5.6.26

- Bug #72907 - null pointer deref, segfault in `gc_remove_zval_from_buffer` (`zend_gc.c:260`).
- Bug #71514 - Bad `dba_replace` condition because of wrong API usage.
- Bug #70825 - Cannot fetch multiple values with group in ini file.
- Bug #72926 - Uninitialized Thumbnail Data Leads To Memory Leakage in `exif_process_IFD_in_TIFF`.
- Bug #70195 - Cannot upload file using `ftp_put` to FTPES with `require_ssl_reuse`.
- Bug #66005 - `imagecopy` does not support 1bit transparency on truecolor images.
- Bug #72913 - `imagecopy()` loses single-color transparency on palette images.
- Bug #68716 - possible resource leaks in `_php_image_convert()`.
- Bug #73007 - add locale length check. (CVE-2016-7416)
- Bug #72787 - `json_decode` reads out of bounds.
- Bug #66797 - `mb_substr` only takes 32-bit signed integer.
- Bug #72910 - Out of bounds heap read in `mbc_to_code()` / triggered by `mb_ereg_match()`.
- Bug #72039 - Use of uninitialised value on `mssql_guid_string`.
- Bug #72293 - Heap overflow in `mysqlnd` related to BIT fields. (CVE-2016-7412)
- Bug #60665 - call to `empty()` on NULL result using `PDO::FETCH_LAZY` returns false.
- Bug #72759 - Regression in `pgo_pgsqL`.
- Bug #72928 - Out of bound when verify signature of zip phar in `phar_parse_zipfile`. (CVE-2016-7414)
- Bug #73035 - Out of bound when verify signature of tar phar in `phar_parse_tarfile`.
- Bug #73029 - Missing type check when unserializing `SplArray`. (CVE-2016-7417)
- Bug #72823 - `strtr` out-of-bound access.
- Bug #72278 - `getimagesize` returning FALSE on valid jpg.
- Bug #65550 - `get_browser()` incorrectly parses entries with "+" sign.
- Bug #71882 - Negative `ftruncate()` on `php://memory` exhausts memory.
- Bug #73011 - integer overflow in `fgets` cause heap corruption.
- Bug #73017 - memory corruption in `wordwrap` function.
- Bug #73045 - integer overflow in `fgetcsv` caused heap corruption.
- Bug #73052 - Memory Corruption in During Deserialized-object Destruction. (CVE-2016-7411)
- Bug #72853 - `stream_set_blocking` doesn't work.
- Bug #72860 - `wddx_deserialize` use-after-free. (CVE-2016-7413)
- Bug #73065 - Out-Of-Bounds Read in `php_wddx_push_element`. (CVE-2016-7418)
- Bug #72085 - SEGV on unknown address `zif_xml_parse`.
- Bug #72927 - integer overflow in `xml_utf8_encode`.
- Bug #68302 - impossible to compile php with zip support.

Prior to 7.0.11

- Bug #72944 - Null pointer deref in `zval_delref_p`.
- Bug #72943 - `assign_dim` on string doesn't reset `hval`.
- Bug #72911 - Memleak in `zend_binary_assign_op_obj_helper`.

- Bug #72813 - Segfault with __get returned by ref.
- Bug #72767 - PHP Segfaults when trying to expand an infinite operator.
- Bug #72854 - PHP Crashes on duplicate destructor call.
- Bug #72857 - stream_socket_recvfrom read access violation.
- Bug #72922 - COM called from PHP does not return out parameters.
- Bug #70825 - Cannot fetch multiple values with group in ini file.
- Bug #70195 - Cannot upload file using ftp_put to FTPES with require_ssl_reuse.
- Bug #72709 - imagesetstyle() causes OOB read for empty \$styles.
- Bug #66005 - imagecopy does not support 1bit transparency on truecolor images.
- Bug #72913 - imagecopy() loses single-color transparency on palette images.
- Bug #68716 - possible resource leaks in _php_image_convert().
- Bug #72320 - iconv_substr returns false for empty strings.
- Bug #72852 - imap_mail null dereference.
- Bug #65732 - grapheme_*() is not Unicode compliant on CR LF sequence.
- Bug #73007 - add locale length check. (CVE-2016-7416)
- Bug #72293 - Heap overflow in mysqlnd related to BIT fields. (CVE-2016-7412)
- Bug #72524 - Binding null values triggers ORA-24816 error.
- Bug #72949 - Typo in opcode error message.
- Bug #72788 - Invalid memory access when using persistent PDO connection.
- Bug #72791 - Memory leak in PDO persistent connection handling.
- Bug #60665 - call to empty() on NULL result using PDO::FETCH_LAZY returns false.
- Bug #72759 - Regression in pgo_pgsqL.
- Bug #72928 - Out of bound when verify signature of zip phar in phar_parse_zipfile. (CVE-2016-7414)
- Bug #73035 - Out of bound when verify signature of tar phar in phar_parse_tarfile.
- Bug #72846 - getConstant for a array constant with constant values returns NULL/NFC/UNKNOWN.
- Bug #72724 - PHP7: session-uploadprogress kills httpd.
- Bug #72940 - SID always return "name=ID", even if session cookie exist.
- Bug #72971 - SimpleXML isset/unset do not respect namespace.
- Bug #72957 - Null coalescing operator doesn't behave as expected with SimpleXMLElement.
- Bug #73029 - Missing type check when unserializing SplArray. (CVE-2016-7417)
- Bug #55451 - substr_compare NULL length interpreted as 0.
- Bug #72278 - getimagesize returning FALSE on valid jpg.
- Bug #65550 - get_browser() incorrectly parses entries with "+" sign.
- Bug #72853 - stream_set_blocking doesn't work.
- Bug #72764 - ftps:// opendir wrapper data channel encryption fails with IIS [FTP 7.5](#), 8.5.
- Bug #71882 - Negative ftruncate() on php://memory exhausts memory.
- Bug #72858 - shm_attach null dereference.
- Bug #72860 - wddx_deserialize use-after-free. (CVE-2016-7413)
- Bug #73065 - Out-Of-Bounds Read in php_wddx_push_element. (CVE-2016-7418)
- Bug #72085 - SEGV on unknown address zif_xml_parse.
- Bug #72714 - _xml_startElementHandler() segmentation fault.
- Bug #68302 - impossible to compile php with zip support.

Successfully exploiting these issues could allow remote attackers to execute arbitrary code in the context of the affected application. Failed exploitation could result in a denial-of-service condition.

RECOMMENDATIONS:

The following actions should be taken:

- Upgrade to the latest version of PHP immediately, after appropriate testing.
- Apply the principle of Least Privilege to all systems and services.
- Verify no unauthorized system modifications have occurred on system before applying patch.
- Remind users not to visit websites or follow links provided by unknown or untrusted sources.
- Do not open email attachments from unknown or untrusted sources.
- Limit user account privileges to only those required.

References:

NOTE: Visiting these links may trigger an IDS signature match for a Possible Encrypted Webshell Download. This is a false positive alert that is matching content on the pages below.

PHP:

<http://www.php.net/ChangeLog-5.php#5.6.26>

<https://secure.php.net/ChangeLog-7.php#7.0.11>

TLP: WHITE

Traffic Light Protocol (TLP): WHITE information may be distributed without restriction, subject to copyright controls.

<http://www.us-cert.gov/tlp/>