

- Secrets
- ABAP
- Apex
- C
- C++
- CloudFormation
- COBOL
- C#
- CSS
- Flex
- Go
- HTML
- Java
- JavaScript
- Kotlin
- Objective C
- PHP
- PL/I
- PL/SQL
- Python
- RPG
- Ruby
- Scala
- Swift
- Terraform
- Text
- TypeScript
- T-SQL
- VB.NET
- VB6
- XML



# PHP static code analysis

Unique rules to find Bugs, Vulnerabilities, Security Hotspots, and Code Smells in your PHP code

All rules 268

Vulnerability 40

Bug 51

Security Hotspot 33

Code Smell 144

Tags

Search by name...



be used to end lines

Code Smell

More than one property should not be declared per statement

Code Smell

The "var" keyword should not be used

Code Smell

"<?php" and "<?=" tags should be used

Code Smell

File names should comply with a naming convention

Code Smell

Comments should not be located at the end of lines of code

Code Smell

Local variable and function parameter names should comply with a naming convention

Code Smell

Field names should comply with a naming convention

Code Smell

Lines should not end with trailing whitespaces

Code Smell

Files should contain an empty newline at the end

Code Smell

Modifiers should be declared in the correct order

Code Smell

An open curly brace should be located at the beginning of a line

## Deprecated features should not be used

Analyze your code

Code Smell Major ? obsolete






Deprecated language features are those that have been retained temporarily for backward compatibility, but which will eventually be removed from the language. In effect, deprecation announces a grace period to allow the smooth transition from the old features to the new ones. In that period, no use of the deprecated features should be added to the code, and all existing uses should be gradually removed.

The following functions were deprecated in PHP 5:

Deprecated	Use Instead
call_user_method()	call_user_func()
call_user_method_array()	call_user_func_array()
define_syslog_variables()	
dl()	
ereg()	preg_match()
ereg_replace()	preg_replace() (note that this is deprecated in PHP 5.5)
eregi()	preg_match() with 'i' modifier
eregi_replace()	preg_replace() with 'i' modifier
set_magic_quotes_runtime() and its alias, magic_quotes_runtime()	
session_register()	\$_SESSION superglobal
session_unregister()	\$_SESSION superglobal
session_is_registered()	\$_SESSION superglobal
set_socket_blocking()	stream_set_blocking()
split()	preg_split()
spliti()	preg_split() with 'i' modifier
sql_regcase()	
mysql_db_query()	mysql_select_db() and mysql_query()
mysql_escape_string()	mysql_real_escape_string()
Passing locale category names as strings	Use the LC_* family of constants

The following functions were deprecated in PHP 7:

Deprecated	Use Instead
__autoload()	spl_autoload_register()
create_function()	anonymous function
parse_str() without second argument	parse_str() with second argument
gmp_random()	gmp_random_bits() or gmp_random_range()
each()	foreach
assert() with string argument	

 Code Smell
An open curly brace should be located at the end of a line
 Code Smell
Tabulation characters should not be used
 Code Smell
Method and function names should comply with a naming convention
 Code Smell
Creating cookies with broadly defined "domain" flags is security-sensitive
 Security Hotspot

Deprecated	Use Instead
Defining case-insensitive constants by calling <code>define()</code> with <code>true</code> as third parameter	<code>define("myconst", \$value)</code> or <code>define("myconst", \$value, false)</code>
<code>FILTER_FLAG_SCHEME_REQUIRED</code> and <code>FILTER_FLAG_HOST_REQUIRED</code> flags	<code>FILTER_VALIDATE_URL</code> flag
<code>fgetss()</code> function, "string.strip_tags" stream filter name, <code>SplFileObject::fgetss()</code> method and <code>gzgetss()</code> function	
<code>mbregex_encoding()</code> , <code>mbereg()</code> , <code>mbereg_i()</code> , <code>mbereg_replace()</code> , <code>mbereg_i_replace()</code> , <code>mbsplit()</code> , <code>mbereg_match()</code> , <code>mbereg_search()</code> , <code>mbereg_search_pos()</code> , <code>mbereg_search_regs()</code> , <code>mbereg_search_init()</code> , <code>mbereg_search_getregs()</code> , <code>mbereg_search_getpos()</code> , <code>mbereg_search_setpos()</code>	Use the corresponding <code>mb_ereg_*()</code> variants instead
string search functions with integer needle ( <code>strstr</code> , <code>strrchr</code> , <code>strstr</code> , <code>stripos</code> , <code>stripos</code> , <code>strrpos</code> , <code>strpos</code> , <code>strchr</code> )	use a string needle instead
<code>image2wbmp()</code>	<code>imagewbmp()</code>
<code>Normalizer::NONE</code>	
Defining an <code>assert()</code> function inside a namespace	use the standard <code>assert()</code> function
<b>See</b> <ul style="list-style-type: none"> <li><a href="#">PHP RFC: Deprecations for PHP 7.3</a></li> <li><a href="#">PHP RFC: Deprecate and Remove Case-Insensitive Constants</a></li> </ul>	
Available In: 