Fortify Security Report

Apr 11, 2023

Administrator

Executive Summary

Issues Overview

On Mar 16, 2023, a source code review was performed over the BD code base. 1,177 files, 272,888 LOC (Executable) were scanned and reviewed for defects that could lead to potential security vulnerabilities. A total of 4911 reviewed findings were uncovered during the analysis.

Issues by Category	
Cross-Site Scripting: Persistent	3932
SQL Injection	711
Password Management: Hardcoded Password	111
Cross-Site Scripting: Reflected	63
Header Manipulation: SMTP	27
Privacy Violation	23
Password Management: Empty Password	13
Cross-Site Scripting: DOM	5
Open Redirect	4
Path Manipulation	3
Privacy Violation: Autocomplete	3
System Information Leak: External	3
Credential Management: Hardcoded API Credentials	2
Dynamic Code Evaluation: Code Injection	2
Dynamic Code Evaluation: Insecure Transport	2
Key Management: Hardcoded Encryption Key	2
Privacy Violation: Shoulder Surfing	2
Dangerous Function	1
Key Management: Empty Encryption Key	1
Weak Encryption	1

Recommendations and Conclusions

The Issues Category section provides Fortify recommendations for addressing issues at a generic level. The recommendations for specific fixes can be extrapolated from those generic recommendations by the development group.

Copyright 2021 Micro Focus or one of its affiliates.

Project Summary

Code Base Summary

Code location: F:/BD

Number of Files: 1177

Lines of Code: 272888

Build Label: <No Build Label>

Scan Information

Scan time: 52:03

SCA Engine version: 21.2.1.0004

Machine Name: SRV156

Username running scan: Administrator

Results Certification

Results Certification Valid

Details:

Results Signature:

SCA Analysis Results has Valid signature

Rules Signature:

There were no custom rules used in this scan

Attack Surface

Attack Surface:

Command Line Arguments:

null.null.null

Environment Variables:

null.null.getenv

File System:

null.null.file

Private Information:

null.null.null

Serialized Data:

null.null.unserialize

Stream:

null.null.fgets

null.null.fread

null.null.stream_get_contents

Copyright 2021 Micro Focus or one of its affiliates.

System Information:
null.null.null
null.null.getcwd
null.null.getmypid
null.null.mysql_error
null.null.xml_error_string
null.simplexmlelement.getnamespaces

Web:
null.null.null
null.~JS_Generic.val

Filter Set Summary

Current Enabled Filter Set:
Quick View

Filter Set Details:

Folder Filters:
If [fortify priority order] contains critical Then set folder to Critical
If [fortify priority order] contains high Then set folder to High
If [fortify priority order] contains medium Then set folder to Medium
If [fortify priority order] contains low Then set folder to Low
Visibility Filters:
If impact is not in range [2.5, 5.0] Then hide issue
If likelihood is not in range (1.0, 5.0] Then hide issue

Audit Guide Summary

Audit guide not enabled

Copyright 2021 Micro Focus or one of its affiliates.

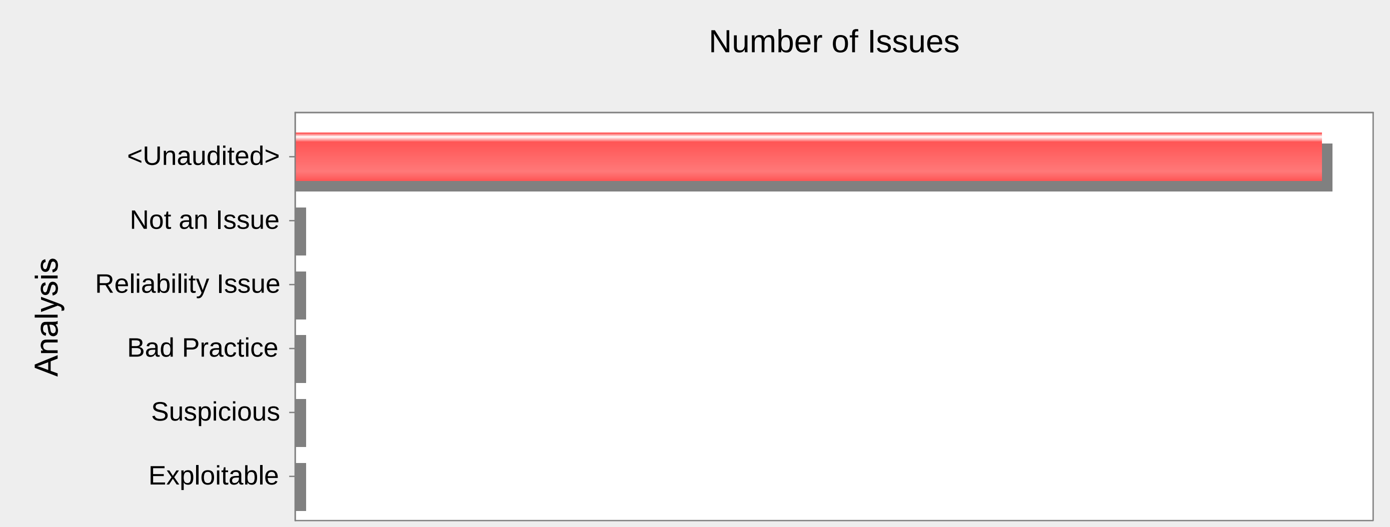
Results Outline

Overall number of results

The scan found 4911 issues.

Vulnerability Examples by Category

Category: Cross-Site Scripting: Persistent (3932 Issues)



Abstract:

Line 122 of index.php sends unvalidated data to a web browser, which can result in the browser executing malicious code.

Explanation:

Cross-site scripting (XSS) vulnerabilities occur when:

1. Data enters a web application through an untrusted source. In the case of persistent (also known as stored) XSS, the untrusted source is typically a database or other back-end data store, while in the case of reflected XSS it is typically a web request.

2. The data is included in dynamic content that is sent to a web user without validation.

The malicious content sent to the web browser often takes the form of a JavaScript segment, but can also include HTML, Flash or any other type of code that the browser executes. The variety of attacks based on XSS is almost limitless, but they commonly include transmitting private data such as cookies or other session information to the attacker, redirecting the victim to web content controlled by the attacker, or performing other malicious operations on the user's machine under the guise of the vulnerable site.

Example 1: The following PHP code segment queries a database for an employee with a given ID and prints the corresponding employee's name.

```
<?php...
$con = mysql_connect($server,$user,$password);
...
$result = mysql_query("select * from emp where id="+eid);
$row = mysql_fetch_array($result)
echo 'Employee name: ', mysql_result($row,0,'name');
...
?>
```

This code functions correctly when the values of name are well-behaved, but it does nothing to prevent exploits if they are not. This code can appear less dangerous because the value of name is read from a database, whose contents are apparently managed by the application. However, if the value of name originates from user-supplied data, then the database can be a conduit for malicious content. Without proper input validation on all data stored in the database, an attacker may execute malicious commands in the user's web browser. This type of exploit, known as Persistent (or Stored) XSS, is particularly insidious because the indirection caused by the data store makes it difficult to identify the threat and increases the possibility that the attack might affect multiple users. XSS got its start in this form with web sites that offered a "guestbook" to visitors. Attackers would include JavaScript in their guestbook entries, and all subsequent visitors to the guestbook page would execute the malicious code.

Example 2: The following PHP code segment reads an employee ID, eid, from an HTTP request and displays it to the user.

```
<?php
```

Copyright 2021 Micro Focus or one of its affiliates.

```
$eid = $_GET['eid'];
...
?>
...
<?php
echo "Employee ID: $eid";
?>
```

As in Example 1, this code operates correctly if eid contains only standard alphanumeric text. If eid has a value that includes metacharacters or source code, then the code is executed by the web browser as it displays the HTTP response.

Initially this might not appear to be much of a vulnerability. After all, why would someone enter a URL that causes malicious code to run on their own computer? The real danger is that an attacker will create the malicious URL, then use email or social engineering tricks to lure victims into visiting a link to the URL. When victims click the link, they unwittingly reflect the malicious content through the vulnerable web application back to their own computers. This mechanism of exploiting vulnerable web applications is known as Reflected XSS.

As the examples demonstrate, XSS vulnerabilities are caused by code that includes unvalidated data in an HTTP response. There are three vectors by which an XSS attack can reach a victim:

- As in Example 1, the application stores dangerous data in a database or other trusted data store. The dangerous data is subsequently read back into the application and included in dynamic content. Persistent XSS exploits occur when an attacker injects dangerous content into a data store that is later read and included in dynamic content. From an attacker's perspective, the optimal place to inject malicious content is in an area that is displayed to either many users or particularly interesting users. Interesting users typically have elevated privileges in the application or interact with sensitive data that is valuable to the attacker. If one of these users executes malicious content, the attacker may be able to perform privileged operations on behalf of the user or gain access to sensitive data belonging to the user.

- As in Example 2, data is read directly from the HTTP request and reflected back in the HTTP response. Reflected XSS exploits occur when an attacker causes a user to supply dangerous content to a vulnerable web application, which is then reflected back to the user and executed by the web browser. The most common mechanism for delivering malicious content is to include it as a parameter in a URL that is posted publicly or emailed directly to victims. URLs constructed in this manner constitute the core of many phishing schemes, whereby an attacker convinces victims to visit a URL that refers to a vulnerable site. After the site reflects the attacker's content back to the user, the content is executed and proceeds to transfer private information, such as cookies that might include session information, from the user's machine to the attacker or perform other nefarious activities.

- A source outside the application stores dangerous data in a database or other data store, and the dangerous data is subsequently read back into the application as trusted data and included in dynamic content.

Recommendations:

The solution to prevent XSS is to ensure that validation occurs in the required places and that relevant properties are set to prevent vulnerabilities.

Because XSS vulnerabilities occur when an application includes malicious data in its output, one logical approach is to validate data immediately before it leaves the application. However, because web applications often have complex and intricate code for generating dynamic content, this method is prone to errors of omission (missing validation). An effective way to mitigate this risk is to also perform input validation for XSS.

Web applications must validate all input to prevent other vulnerabilities, such as SQL injection, so augmenting an application's existing input validation mechanism to include checks for XSS is generally relatively easy. Despite its value, input validation for XSS does not take the place of rigorous output validation. An application might accept input through a shared data store or other trusted source, and that data store might accept input from a source that does not perform adequate input validation. Therefore, the application cannot implicitly rely on the safety of this or any other data. This means that the best way to prevent XSS vulnerabilities is to validate everything that enters the application and leaves the application destined for the user.

The most secure approach to validation for XSS is to create an allow list of safe characters that can appear in HTTP content and accept input composed exclusively of characters in the approved set. For example, a valid username might only include alphanumeric characters or a phone number might only include digits 0-9. However, this solution is often infeasible in web applications because many characters that have special meaning to the browser must be considered valid input after they are encoded, such as a web design bulletin board that must accept HTML fragments from its users.

A more flexible, but less secure approach is to implement a deny list, which selectively rejects or escapes potentially dangerous characters before using the input. To form such a list, you first need to understand the set of characters that hold special meaning for web browsers. Although the HTML standard defines which characters have special meaning, many web browsers try to correct common mistakes in HTML and might treat other characters as special in certain contexts. This is why we do not recommend the use of deny lists as a means to prevent XSS. The CERT(R) Coordination Center at the Software Engineering Institute at Carnegie Mellon University provides the following details about special characters in various contexts [1]:

In the content of a block-level element (in the middle of a paragraph of text):

- "<" is special because it introduces a tag.

- "&" is special because it introduces a character entity.

- ">" is special because some browsers treat it as special, on the assumption that the author of the page intended to include an opening "<", but omitted it in error.

Copyright 2021 Micro Focus or one of its affiliates.

The following principles apply to attribute values:

- In attribute values enclosed in double quotes, the double quotes are special because they mark the end of the attribute value.

- In attribute values enclosed in single quotes, the single quotes are special because they mark the end of the attribute value.

- In attribute values without any quotes, white-space characters, such as space and tab, are special.

- "&" is special when used with certain attributes, because it introduces a character entity.

In URLs, for example, a search engine might provide a link within the results page that the user can click to re-run the search. This can be implemented by encoding the search query inside the URL, which introduces additional special characters:

- Space, tab, and new line are special because they mark the end of the URL.

- "&" is special because it either introduces a character entity or separates CGI parameters.

- Non-ASCII characters (that is, everything greater than 127 in the ISO-8859-1 encoding) are not allowed in URLs, so they are considered to be special in this context.

- The "%" symbol must be filtered from input anywhere parameters encoded with HTTP escape sequences are decoded by server-side code. For example, "%" must be filtered if input such as "%68%65%6C%6C%6F" becomes "hello" when it appears on the web page.

Within the body of a <SCRIPT> </SCRIPT>:

- Semicolons, parentheses, curly braces, and new line characters must be filtered out in situations where text could be inserted directly into a pre-existing script tag.

Server-side scripts:

- Server-side scripts that convert any exclamation characters (!) in input to double-quote characters (") on output might require additional filtering.

Other possibilities:

- If an attacker submits a request in UTF-7, the special character '<' appears as '+ADw-' and might bypass filtering. If the output is included in a page that does not explicitly specify an encoding format, then some browsers try to intelligently identify the encoding based on the content (in this case, UTF-7).

After you identify the correct points in an application to perform validation for XSS attacks and what special characters the validation should consider, the next challenge is to identify how your validation handles special characters. If special characters are not considered valid input to the application, then you can reject any input that contains special characters as invalid. A second option is to remove special characters with filtering. However, filtering has the side effect of changing any visual representation of the filtered content and might be unacceptable in circumstances where the integrity of the input must be preserved for display.

If input containing special characters must be accepted and displayed accurately, validation must encode any special characters to remove their significance. A complete list of ISO 8859-1 encoded values for special characters is provided as part of the official HTML specification [2].

Many application servers attempt to limit an application's exposure to cross-site scripting vulnerabilities by providing implementations for the functions responsible for setting certain specific HTTP response content that perform validation for the characters essential to a cross-site scripting attack. Do not rely on the server running your application to make it secure. For any developed application, there are no guarantees about which application servers it will run on during its lifetime. As standards and known exploits evolve, there are no guarantees that application servers will continue to stay in sync.

Tips:

1. The Fortify Secure Coding Rulepacks warn about SQL Injection and Access Control: Database issues when untrusted data is written to a database and also treat the database as a source of untrusted data, which can lead to XSS vulnerabilities. If the database is a trusted resource in your environment, use custom filters to filter out dataflow issues that include the DATABASE taint flag or originate from database sources. Nonetheless, it is often still a good idea to validate everything read from the database.

2. Even though URL encoding untrusted data protects against many XSS attacks, some browsers (specifically, Internet Explorer 6 and 7 and possibly others) automatically decode content at certain locations within the Document Object Model (DOM) prior to passing it to the JavaScript interpreter. To reflect this danger, the Rulepacks no longer treat URL encoding routines as sufficient to protect against cross-site scripting. Data values that are URL encoded and subsequently output will cause Fortify to report Cross-Site Scripting: Poor Validation vulnerabilities.

3. Due to the dynamic nature of PHP, you may see a large number of findings in PHP library files. Consider using a filter file to hide specific findings from view. For instructions on creating a filter file, see Advanced Options in the Fortify Static Code Analyzer User Guide.

index.php, line 122 (Cross-Site Scripting: Persistent)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Input Validation and Representation		

Copyright 2021 Micro Focus or one of its affiliates.

Abstract: Line 122 of index.php sends unvalidated data to a web browser, which can result in the browser executing malicious code.

Source: index.php:113 mysql_db_query()

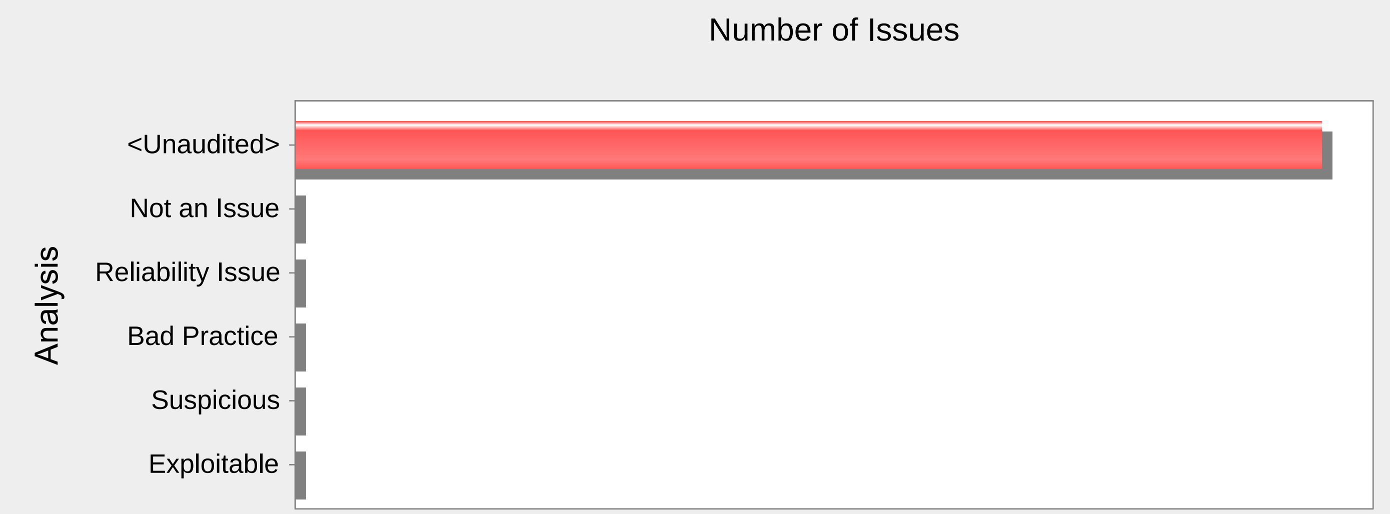
```
111      if(@$_REQUEST['suboperation']=='download_villages'){// Ýêñïîðò ñïèñêà ñ¸ë èç áàçû
112        $Query="SELECT net5.password,net5.name,net5.ip,net5.ip_del_time FROM net5 LEFT
JOIN net5 ON net5.reg_id=net5.id ORDER BY net5.password,net5.password";
113        $res=@mysql_db_query('host_29723_mirbase2',$Query);
114        $mime_type = ($UserBrowser == 'IE' || $UserBrowser == 'Opera') ?
'application/octetstream' : 'application/octet-stream';
115        //@ob_end_clean(); /// decrease cpu usage extreme
```

Sink: index.php:122 builtin_echo()

```
120        for($i=0;$i<mysql_num_rows($res);$i++){// Âûâîä ñ¸ë â ôàéë
121        $A=@mysql_fetch_row($res);
122        echo $A[0]."\t".$A[1]."\t".$A[2]."\t".$A[3]."\r\n";flush();
123        }// Âûâîä ñ¸ë â ôàéë
124        flush();die();
```

Copyright 2021 Micro Focus or one of its affiliates.

Category: SQL Injection (711 Issues)



Abstract:

Line 67 of 1.php invokes a SQL query built with input that comes from an untrusted source. This call could allow an attacker to modify the statement's meaning or to execute arbitrary SQL commands.

Explanation:

SQL injection errors occur when:

1. Data enters a program from an untrusted source.

2. The data is used to dynamically construct a SQL query.

Example 1: The following code dynamically constructs and executes a SQL query that searches for items matching a specified name. The query restricts the items displayed to those where the owner matches the user name of the currently-authenticated user.

```
...
$userName = $_SESSION['userName'];
$itemName = $_POST['itemName'];
$query = "SELECT * FROM items WHERE owner = '$userName' AND itemname = '$itemName';";
$result = mysql_query($query);
...
```

The query intends to execute the following code:

```
SELECT * FROM items
WHERE owner = <userName>
AND itemname = <itemName>;
```

However, because the query is constructed dynamically by concatenating a constant query string and a user input string, the query only behaves correctly if itemName does not contain a single-quote character. If an attacker with the user name wiley enters the string "name' OR 'a'='a" for itemName, then the query becomes the following:

```
SELECT * FROM items
WHERE owner = 'wiley'
AND itemname = 'name' OR 'a'='a';
```

The addition of the OR 'a'='a' condition causes the where clause to always evaluate to true, so the query becomes logically equivalent to the much simpler query:

```
SELECT * FROM items;
```

This simplification of the query allows the attacker to bypass the requirement that the query must only return items owned by the authenticated user. The query now returns all entries stored in the items table, regardless of their specified owner.

Example 2: This example examines the effects of a different malicious value passed to the query constructed and executed in Example 1. If an attacker with the user name wiley enters the string "name'; DELETE FROM items; --" for itemName, then the query becomes the following two queries:

Copyright 2021 Micro Focus or one of its affiliates.

SELECT * FROM items

WHERE owner = 'wiley'

AND itemname = 'name';

DELETE FROM items;

--'

Many database servers, including Microsoft(R) SQL Server 2000, allow multiple SQL statements separated by semicolons to be executed at once. While this attack string results in an error on Oracle and other database servers that do not allow the batch-execution of statements separated by semicolons, on databases that do allow batch execution, this type of attack allows the attacker to execute arbitrary commands against the database.

Notice the trailing pair of hyphens (--), which specifies to most database servers that the remainder of the statement is to be treated as a comment and not executed [4]. In this case the comment character serves to remove the trailing single-quote left over from the modified query. On a database where comments are not allowed to be used in this way, the general attack could still be made effective using a trick similar to the one shown in Example 1. If an attacker enters the string "name'); DELETE FROM items; SELECT * FROM items WHERE 'a'='a", the following three valid statements will be created:

SELECT * FROM items

WHERE owner = 'wiley'

AND itemname = 'name';

DELETE FROM items;

SELECT * FROM items WHERE 'a'='a';

One traditional approach to preventing SQL injection attacks is to handle them as an input validation problem and either accept only characters from an allow list of safe values or identify and escape a list of potentially malicious values (deny list). Checking an allow list can be a very effective means of enforcing strict input validation rules, but parameterized SQL statements require less maintenance and can offer more guarantees with respect to security. As is almost always the case, implementing a deny list is riddled with loopholes that make it ineffective at preventing SQL injection attacks. For example, attackers may:

- Target fields that are not quoted
- Find ways to bypass the need for certain escaped metacharacters
- Use stored procedures to hide the injected metacharacters

Manually escaping characters in input to SQL queries can help, but it will not make your application secure from SQL injection attacks.

Another solution commonly proposed for dealing with SQL injection attacks is to use stored procedures. Although stored procedures prevent some types of SQL injection attacks, they fail to protect against many others. Stored procedures typically help prevent SQL injection attacks by limiting the types of statements that can be passed to their parameters. However, there are many ways around the limitations and many interesting statements that can still be passed to stored procedures. Again, stored procedures can prevent some exploits, but they will not make your application secure against SQL injection attacks.

Recommendations:

The root cause of a SQL injection vulnerability is the ability of an attacker to change context in the SQL query, causing a value that the programmer intended to be interpreted as data to be interpreted as a command instead. When a SQL query is constructed, the programmer knows what should be interpreted as part of the command and what should be interpreted as data. Parameterized SQL statements can enforce this behavior by disallowing data-directed context changes and preventing nearly all SQL injection attacks. Parameterized SQL statements are constructed using strings of regular SQL, but where user-supplied data needs to be included, they include bind parameters, which are placeholders for data that is subsequently inserted. In other words, bind parameters allow the programmer to explicitly specify to the database what should be treated as a command and what should be treated as data. When the program is ready to execute a statement, it specifies to the database the runtime values to use for each of the bind parameters without the risk that the data will be interpreted as a modification to the command.

When connecting to MySQL, the previous example can be rewritten to use parameterized SQL statements (instead of concatenating user supplied strings) as follows:

```
...
$mysqli = new mysqli($host,$dbuser, $dbpass, $db);
$userName = $_SESSION['userName'];
$itemName = $_POST['itemName'];
$query = "SELECT * FROM items WHERE owner = ? AND itemname = ?";
$stmt = $mysqli->prepare($query);
$stmt->bind_param('ss',$username,$itemName);
$stmt->execute();
...
```

Copyright 2021 Micro Focus or one of its affiliates.

The MySQL Improved extension (mysqli) is available for PHP5 users of MySQL. Code that relies on a different database should check for similar extensions.

More complicated scenarios, often found in report generation code, require that user input affect the structure of the SQL statement, for instance by adding a dynamic constraint in the WHERE clause. Do not use this requirement to justify concatenating user input to create a query string. Prevent SQL injection attacks where user input must affect command structure with a level of indirection: create a set of legitimate strings that correspond to different elements you might include in a SQL statement. When constructing a statement, use input from the user to select from this set of application-controlled values.

Tips:

1. A common mistake is to use parameterized SQL statements that are constructed by concatenating user-controlled strings. Of course, this defeats the purpose of using parameterized SQL statements. If you are not certain that the strings used to form parameterized statements are constants controlled by the application, do not assume that they are safe because they are not being executed directly as SQL strings. Thoroughly investigate all uses of user-controlled strings in SQL statements and verify that none can be used to modify the meaning of the query.

2. Due to the dynamic nature of PHP, you may see a large number of findings in PHP library files. Consider using a filter file to hide specific findings from view. For instructions on creating a filter file, see Advanced Options in the Fortify Static Code Analyzer User Guide.

1.php, line 67 (SQL Injection)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Input Validation and Representation		

Abstract: Line 67 of 1.php invokes a SQL query built with input that comes from an untrusted source. This call could allow an attacker to modify the statement's meaning or to execute arbitrary SQL commands.

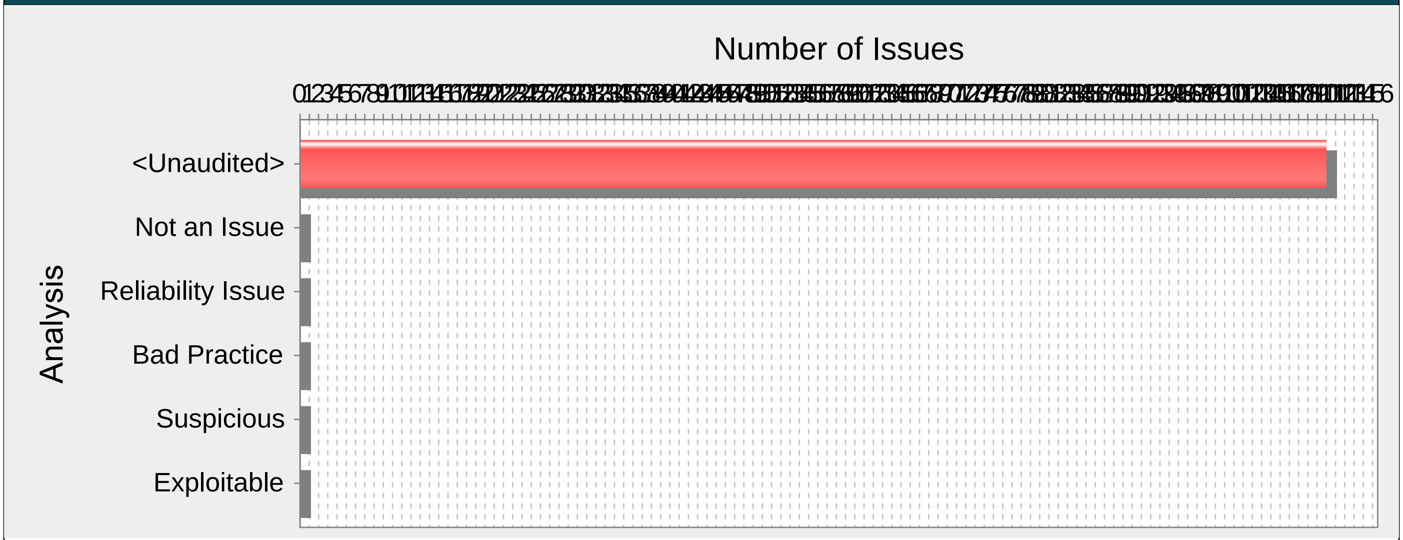
Source: 1.php:58 Read $_POST['opis4']()

```
56   $opis2 =htmlspecialchars ($_POST['opis2']);
57   $opis3 =htmlspecialchars ($_POST['opis3']);
58   $opis4 =htmlspecialchars ($_POST['opis4']);
59   $opis5 =htmlspecialchars ($_POST['opis5']);
60   $opis6 =htmlspecialchars ($_POST['opis6']);
```

Sink: 1.php:67 mysql_query()

```
65
66
67    $query = mysql_query("INSERT INTO
audit(Audittype,opis11,nomerotcheta,auditna,naimenovanieotcheta,reshenieprav lenie,dataotcheta,narushenie,meropiatia,otvetctvenoepodrazdel,auditori,liniacompitenci,reiting,pervonachalnidataispolenie,status,novaiadataispolnenia,monitoringperiod,rassilka,eladressrassilki,komment,neophodimiecorrectmeri,opis1,opis2,opis3,opis4,opis5,opis6,opis7,opis8)
values('$Audittype','$opis11','$nomerotcheta','$auditna','$naimenovanieotcheta','$reshenieprav lenie','$dataotcheta',...
68
69      mysql_close();
```

Category: Password Management: Hardcoded Password (111 Issues)



Abstract:

Hardcoded passwords could compromise system security in a way that cannot be easily remedied.

Explanation:

It is never a good idea to hardcode a password. Not only does hardcoding a password allow all of the project's developers to view the password, it also makes fixing the problem extremely difficult. After the code is in production, the password cannot be changed without patching the software. If the account protected by the password is compromised, the owners of the system must choose between security and availability.

Example: The following code uses a hardcoded password to connect to a database:

```
...
$link = mysql_connect($url, 'scott', 'tiger');
if (!$link) {
die('Could not connect: ' . mysql_error());
}
...
```

This code will run successfully, but anyone who has access to it will have access to the password. After the program ships, there is likely no way to change the database user "scott" with a password of "tiger" unless the program is patched. An employee with access to this information can use it to break into the system.

Recommendations:

Passwords should never be hardcoded and should generally be obfuscated and managed in an external source. Storing passwords in plain text anywhere on the system allows anyone with sufficient permissions to read and potentially misuse the password.

Some third-party products claim the ability to manage passwords in a more secure way. For a secure solution, the only viable option today appears to be a proprietary one that you create.

Tips:

1. To identify null, empty, or hardcoded passwords, default rules only consider fields and variables that contain the word password. However, the Fortify Custom Rules Editor provides the Password Management wizard that makes it easy to create rules for detecting password management issues on custom-named fields and variables.

2. Due to the dynamic nature of PHP, you may see a large number of findings in PHP library files. Consider using a filter file to hide specific findings from view. For instructions on creating a filter file, see Advanced Options in the Fortify Static Code Analyzer User Guide.

1.php, line 27 (Password Management: Hardcoded Password)

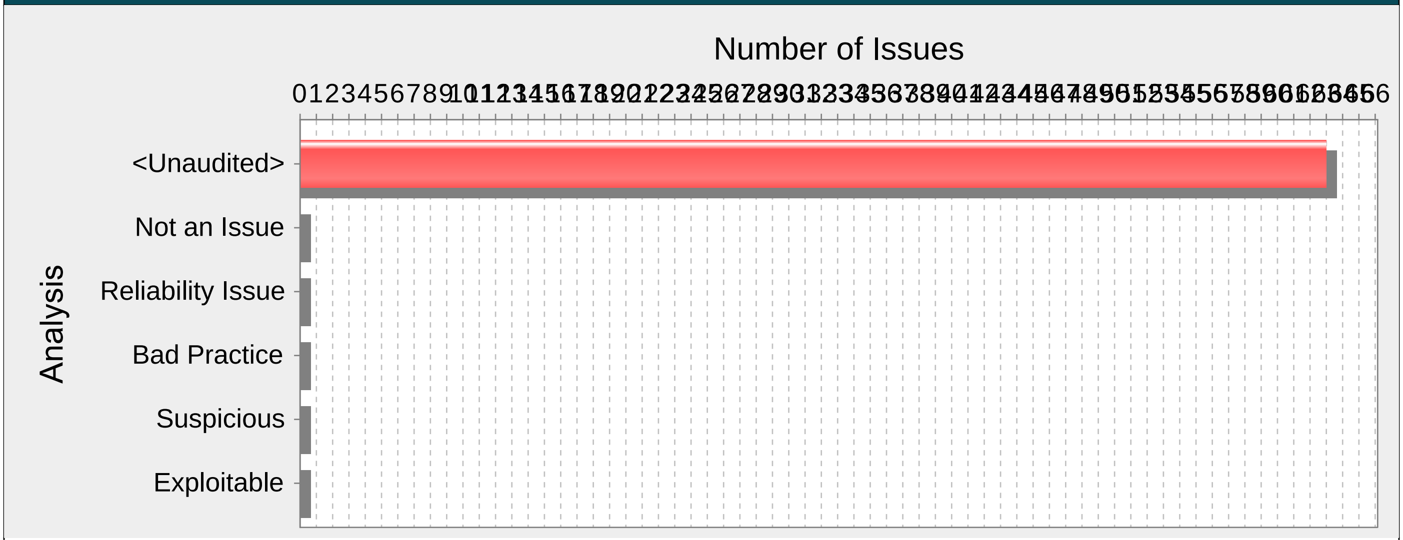
Fortify Priority:	High	**Folder**	High
Kingdom:	Security Features		
Abstract:	Hardcoded passwords could compromise system security in a way that cannot be easily remedied.		
Sink:	1.php:27 mysql_connect()		

```
25 <?php
26
27 $connect=@mysql_connect('localhost:3306','root','rFn76#bL3') or die(mysql_error());
28  mysql_select_db('host_29723_mirbase');
```

Copyright 2021 Micro Focus or one of its affiliates.

Copyright 2021 Micro Focus or one of its affiliates.

Category: Cross-Site Scripting: Reflected (63 Issues)



Abstract:

Line 73 of index.php sends unvalidated data to a web browser, which can result in the browser executing malicious code.

Explanation:

Cross-site scripting (XSS) vulnerabilities occur when:

1. Data enters a web application through an untrusted source. In the case of reflected XSS, the untrusted source is typically a web request, while in the case of persisted (also known as stored) XSS it is typically a database or other back-end data store.

2. The data is included in dynamic content that is sent to a web user without validation.

The malicious content sent to the web browser often takes the form of a JavaScript segment, but can also include HTML, Flash or any other type of code that the browser executes. The variety of attacks based on XSS is almost limitless, but they commonly include transmitting private data such as cookies or other session information to the attacker, redirecting the victim to web content controlled by the attacker, or performing other malicious operations on the user's machine under the guise of the vulnerable site.

Example 1: The following PHP code segment reads an employee ID, eid, from an HTTP request and displays it to the user.

```
<?php
$eid = $_GET['eid'];
...
?>
...
<?php
echo "Employee ID: $eid";
?>
```

The code in this example operates correctly if eid contains only standard alphanumeric text. If eid has a value that includes metacharacters or source code, then the code is executed by the web browser as it displays the HTTP response.

Initially this might not appear to be much of a vulnerability. After all, why would someone enter a URL that causes malicious code to run on their own computer? The real danger is that an attacker will create the malicious URL, then use email or social engineering tricks to lure victims into visiting a link to the URL. When victims click the link, they unwittingly reflect the malicious content through the vulnerable web application back to their own computers. This mechanism of exploiting vulnerable web applications is known as Reflected XSS.

Example 2: The following PHP code segment queries a database for an employee with a given ID and prints the corresponding employee's name.

```
<?php...
$con = mysql_connect($server,$user,$password);
...
$result = mysql_query("select * from emp where id="+eid);
$row = mysql_fetch_array($result)
echo 'Employee name: ', mysql_result($row,0,'name');
...
?>
```

Copyright 2021 Micro Focus or one of its affiliates.

As in Example 1, this code functions correctly when the values of name are well-behaved, but it does nothing to prevent exploits if they are not. Again, this code can appear less dangerous because the value of name is read from a database, whose contents are apparently managed by the application. However, if the value of name originates from user-supplied data, then the database can be a conduit for malicious content. Without proper input validation on all data stored in the database, an attacker may execute malicious commands in the user's web browser. This type of exploit, known as Persistent (or Stored) XSS, is particularly insidious because the indirection caused by the data store makes it difficult to identify the threat and increases the possibility that the attack might affect multiple users. XSS got its start in this form with web sites that offered a "guestbook" to visitors. Attackers would include JavaScript in their guestbook entries, and all subsequent visitors to the guestbook page would execute the malicious code.

As the examples demonstrate, XSS vulnerabilities are caused by code that includes unvalidated data in an HTTP response. There are three vectors by which an XSS attack can reach a victim:

- As in Example 1, data is read directly from the HTTP request and reflected back in the HTTP response. Reflected XSS exploits occur when an attacker causes a user to supply dangerous content to a vulnerable web application, which is then reflected back to the user and executed by the web browser. The most common mechanism for delivering malicious content is to include it as a parameter in a URL that is posted publicly or emailed directly to victims. URLs constructed in this manner constitute the core of many phishing schemes, whereby an attacker convinces victims to visit a URL that refers to a vulnerable site. After the site reflects the attacker's content back to the user, the content is executed and proceeds to transfer private information, such as cookies that might include session information, from the user's machine to the attacker or perform other nefarious activities.

- As in Example 2, the application stores dangerous data in a database or other trusted data store. The dangerous data is subsequently read back into the application and included in dynamic content. Persistent XSS exploits occur when an attacker injects dangerous content into a data store that is later read and included in dynamic content. From an attacker's perspective, the optimal place to inject malicious content is in an area that is displayed to either many users or particularly interesting users. Interesting users typically have elevated privileges in the application or interact with sensitive data that is valuable to the attacker. If one of these users executes malicious content, the attacker may be able to perform privileged operations on behalf of the user or gain access to sensitive data belonging to the user.

- A source outside the application stores dangerous data in a database or other data store, and the dangerous data is subsequently read back into the application as trusted data and included in dynamic content.

Recommendations:

The solution to prevent XSS is to ensure that validation occurs in the required places and that relevant properties are set to prevent vulnerabilities.

Because XSS vulnerabilities occur when an application includes malicious data in its output, one logical approach is to validate data immediately before it leaves the application. However, because web applications often have complex and intricate code for generating dynamic content, this method is prone to errors of omission (missing validation). An effective way to mitigate this risk is to also perform input validation for XSS.

Web applications must validate all input to prevent other vulnerabilities, such as SQL injection, so augmenting an application's existing input validation mechanism to include checks for XSS is generally relatively easy. Despite its value, input validation for XSS does not take the place of rigorous output validation. An application might accept input through a shared data store or other trusted source, and that data store might accept input from a source that does not perform adequate input validation. Therefore, the application cannot implicitly rely on the safety of this or any other data. This means that the best way to prevent XSS vulnerabilities is to validate everything that enters the application and leaves the application destined for the user.

The most secure approach to validation for XSS is to create an allow list of safe characters that can appear in HTTP content and accept input composed exclusively of characters in the approved set. For example, a valid username might only include alphanumeric characters or a phone number might only include digits 0-9. However, this solution is often infeasible in web applications because many characters that have special meaning to the browser must be considered valid input after they are encoded, such as a web design bulletin board that must accept HTML fragments from its users.

A more flexible, but less secure approach is to implement a deny list, which selectively rejects or escapes potentially dangerous characters before using the input. To form such a list, you first need to understand the set of characters that hold special meaning for web browsers. Although the HTML standard defines which characters have special meaning, many web browsers try to correct common mistakes in HTML and might treat other characters as special in certain contexts. This is why we do not recommend the use of deny lists as a means to prevent XSS. The CERT(R) Coordination Center at the Software Engineering Institute at Carnegie Mellon University provides the following details about special characters in various contexts [1]:

In the content of a block-level element (in the middle of a paragraph of text):

- "<" is special because it introduces a tag.

- "&" is special because it introduces a character entity.

- ">" is special because some browsers treat it as special, on the assumption that the author of the page intended to include an opening "<", but omitted it in error.

The following principles apply to attribute values:

- In attribute values enclosed in double quotes, the double quotes are special because they mark the end of the attribute value.

- In attribute values enclosed in single quotes, the single quotes are special because they mark the end of the attribute value.

- In attribute values without any quotes, white-space characters, such as space and tab, are special.

- "&" is special when used with certain attributes, because it introduces a character entity.

In URLs, for example, a search engine might provide a link within the results page that the user can click to re-run the search. This can be implemented by encoding the search query inside the URL, which introduces additional special characters:

- Space, tab, and new line are special because they mark the end of the URL.

- "&" is special because it either introduces a character entity or separates CGI parameters.

- Non-ASCII characters (that is, everything greater than 127 in the ISO-8859-1 encoding) are not allowed in URLs, so they are considered to be special in this context.

- The "%" symbol must be filtered from input anywhere parameters encoded with HTTP escape sequences are decoded by server-side code. For example, "%" must be filtered if input such as "%68%65%6C%6C%6F" becomes "hello" when it appears on the web page.

Within the body of a <SCRIPT> </SCRIPT>:

- Semicolons, parentheses, curly braces, and new line characters must be filtered out in situations where text could be inserted directly into a pre-existing script tag.

Server-side scripts:

- Server-side scripts that convert any exclamation characters (!) in input to double-quote characters (") on output might require additional filtering.

Other possibilities:

- If an attacker submits a request in UTF-7, the special character '<' appears as '+ADw-' and might bypass filtering. If the output is included in a page that does not explicitly specify an encoding format, then some browsers try to intelligently identify the encoding based on the content (in this case, UTF-7).

After you identify the correct points in an application to perform validation for XSS attacks and what special characters the validation should consider, the next challenge is to identify how your validation handles special characters. If special characters are not considered valid input to the application, then you can reject any input that contains special characters as invalid. A second option is to remove special characters with filtering. However, filtering has the side effect of changing any visual representation of the filtered content and might be unacceptable in circumstances where the integrity of the input must be preserved for display.

If input containing special characters must be accepted and displayed accurately, validation must encode any special characters to remove their significance. A complete list of ISO 8859-1 encoded values for special characters is provided as part of the official HTML specification [2].

Many application servers attempt to limit an application's exposure to cross-site scripting vulnerabilities by providing implementations for the functions responsible for setting certain specific HTTP response content that perform validation for the characters essential to a cross-site scripting attack. Do not rely on the server running your application to make it secure. For any developed application, there are no guarantees about which application servers it will run on during its lifetime. As standards and known exploits evolve, there are no guarantees that application servers will continue to stay in sync.

Tips:

1. The Fortify Secure Coding Rulepacks warn about SQL Injection and Access Control: Database issues when untrusted data is written to a database and also treat the database as a source of untrusted data, which can lead to XSS vulnerabilities. If the database is a trusted resource in your environment, use custom filters to filter out dataflow issues that include the DATABASE taint flag or originate from database sources. Nonetheless, it is often still a good idea to validate everything read from the database.

2. Even though URL encoding untrusted data protects against many XSS attacks, some browsers (specifically, Internet Explorer 6 and 7 and possibly others) automatically decode content at certain locations within the Document Object Model (DOM) prior to passing it to the JavaScript interpreter. To reflect this danger, the Rulepacks no longer treat URL encoding routines as sufficient to protect against cross-site scripting. Data values that are URL encoded and subsequently output will cause Fortify to report Cross-Site Scripting: Poor Validation vulnerabilities.

3. Due to the dynamic nature of PHP, you may see a large number of findings in PHP library files. Consider using a filter file to hide specific findings from view. For instructions on creating a filter file, see Advanced Options in the Fortify Static Code Analyzer User Guide.

index.php, line 73 (Cross-Site Scripting: Reflected)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Input Validation and Representation		
Abstract:	Line 73 of index.php sends unvalidated data to a web browser, which can result in the browser executing malicious code.		
Source:	index.php:73 Read $_SERVER['PHP_SELF']()		

```
71
72        if($oper_id){// --- Äîñòóï çàïðåùåí!!!
73        echo '<form method="POST" action="'.$_SERVER['PHP_SELF'].'"><table width="200"
      align="center">';
74      echo '<table width="200" align="center">';
```

Copyright 2021 Micro Focus or one of its affiliates.

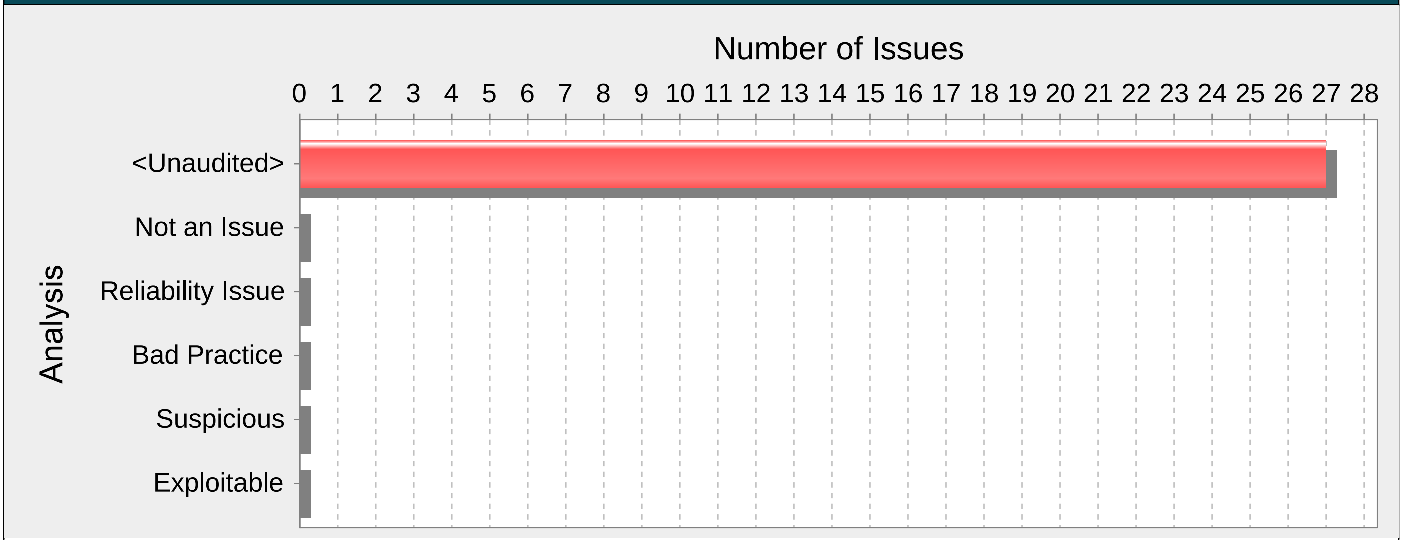
```
75    if(strlen($error_msg)>2)echo '<tr><td align="center"><font
      color="red">'.$error_msg.'</font></td></tr>';
```

Sink: index.php:73 builtin_echo()

```
71
72     if($oper_id){// --- Äîñòóï çàïðåùåí!!!
73     echo '<form method="POST" action="'.$_SERVER['PHP_SELF'].'"><table width="200"
      align="center">';
74    echo '<table width="200" align="center">';
75    if(strlen($error_msg)>2)echo '<tr><td align="center"><font
      color="red">'.$error_msg.'</font></td></tr>';
```

Copyright 2021 Micro Focus or one of its affiliates.

Category: Header Manipulation: SMTP (27 Issues)



Abstract:

Including unvalidated data in an SMTP header can enable attackers to add arbitrary headers, such as CC or BCC that they can use to leak the mail contents to themselves or use the mail server as a spam bot.

Explanation:

SMTP Header Manipulation vulnerabilities occur when:

1. Data enters an application through an untrusted source, most frequently an HTTP request in a web application.

2. The data is included in an SMTP header sent to a mail server without being validated.

As with many software security vulnerabilities, SMTP Header Manipulation is a means to an end, not an end in itself. At its root, the vulnerability is straightforward: an attacker passes malicious data to a vulnerable application, and the application includes the data in an SMTP header.

One of the most common SMTP Header Manipulation attacks is for the use of distributing spam emails. If an application contains a vulnerable "Contact us" form that allows setting the subject and the body of the email, an attacker will be able to set any arbitrary content and inject a CC header with a list of email addresses to spam anonymously since the email will be sent from the victim server.

Example: The following code segment reads the subject and body of a "Contact us" form:

```
$subject = $_GET['subject'];
$body =  $_GET['body'];
mail("support@acme.com", "[Contact us query] " . $subject, $body);
```

Assuming a string consisting of standard alphanumeric characters, such as "Page not working" is submitted in the request, the SMTP headers might take the following form:

```
...
subject: [Contact us query] Page not working
...
```

However, because the value of the header is constructed from unvalidated user input the response will only maintain this form if the value submitted for subject does not contain any CR and LF characters. If an attacker submits a malicious string, such as "Congratulations!! You won the lottery!!!\r\ncc:victim1@mail.com,victim2@mail.com ...", then the SMTP headers would be of the following form:

```
...
subject: [Contact us query] Congratulations!! You won the lottery
cc: victim1@mail.com,victim2@mail.com
...
```

This will effectively allow an attacker to craft spam messages or to send anonymous emails amongst other attacks.

Recommendations:

The solution to prevent SMTP Header Manipulation is to ensure that input validation occurs in the required places and checks for the correct properties.

Copyright 2021 Micro Focus or one of its affiliates.

Since SMTP Header Manipulation vulnerabilities occur when an application includes malicious data in its output, one logical approach is to validate data immediately before it is used in the header context and make sure there are no illegal CRLF characters that can break the header structure.

send - Copy.php, line 1559 (Header Manipulation: SMTP)

Fortify Priority:	High	**Folder**	High
Kingdom:	Input Validation and Representation		

Abstract: Including unvalidated data in an SMTP header can enable attackers to add arbitrary headers, such as CC or BCC that they can use to leak the mail contents to themselves or use the mail server as a spam bot.

Source: send - Copy.php:1520 Read $_POST['email1']()

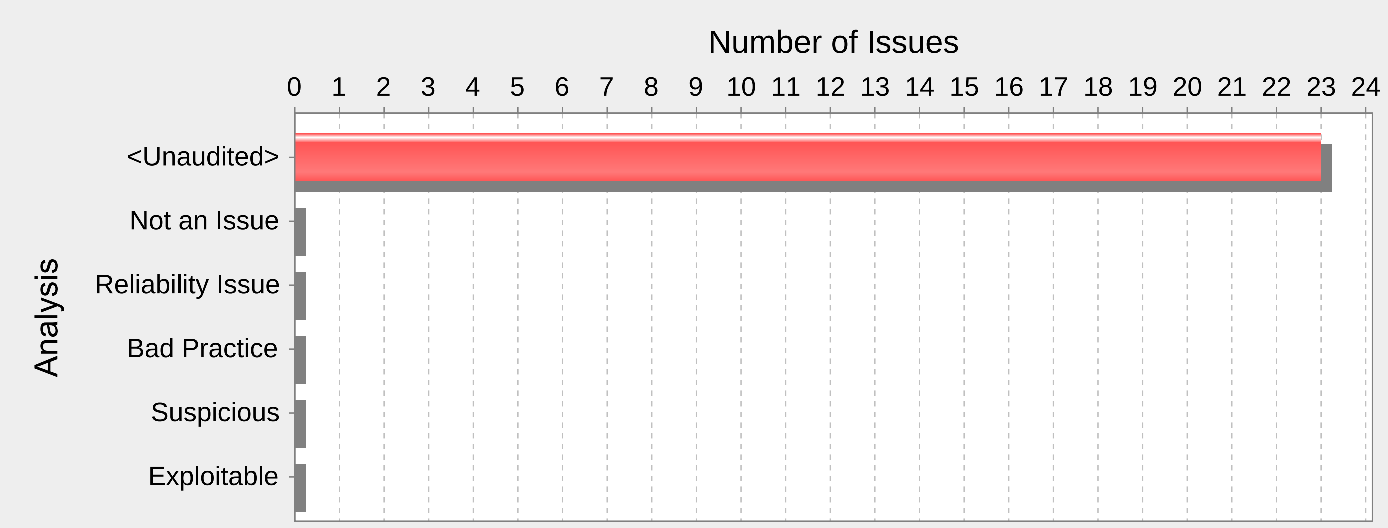
```
1518    </p><?php error_reporting(0);
1519    if (isset($_POST['email'])) {$email = $_POST['email'];}
1520    if (isset($_POST['email1'])) {$email1 = $_POST['email1'];}
1521    if (isset($_POST['mess'])) {$mess = $_POST['mess'];}
1522    if (isset($_POST['newName222'])) {$mess111 = $_POST['newName222'];}
```

Sink: send - Copy.php:1559 mail()

```
1557    $message = "$mess $mess1  $mess6  $mess2<tr> <td><br> <b> <br/> :</b> $mess3 $mess5
        <br/><b>:</b>  $mess7  <br/><b>: </b>$mess8 <br/><b> :</b> $mess9 <br/><b>  : </b>
        $mess10 <br/><b>   </b>$mess11 <br/> $mess12 <br/> $mess4</td></tr></tr> ";
1558
1559    $send = mail ($to, $subject, $message, $headers);
1560    if ($send == 'true')
1561    {
```

Copyright 2021 Micro Focus or one of its affiliates.

Category: Privacy Violation (23 Issues)



Abstract:

The file class.pop3.php mishandles confidential information on line 334, which can compromise user privacy and is often illegal.

Explanation:

Privacy violations occur when:

1. Private user information enters the program.

2. The data is written to an external location, such as the console, file system, or network.

Example 1: The following code contains a logging statement that tracks the contents of records added to a database by storing them in a log file. Among other values that are stored is the return value from the getPassword() function that returns user-supplied plain text password associated with the account.

```
<?php
$pass = getPassword();
trigger_error($id . ":" . $pass . ":" . $type . ":" . $tstamp);
?>
```

The code in Example 1 logs a plain text password to the application eventlog. Although many developers trust the eventlog as a safe storage location for data, it should not be trusted implicitly, particularly when privacy is a concern.

Private data can enter a program in a variety of ways:

- Directly from the user in the form of a password or personal information

- Accessed from a database or other data store by the application

- Indirectly from a partner or other third party

Sometimes data that is not labeled as private can have a privacy implication in a different context. For example, student identification numbers are usually not considered private because there is no explicit and publicly-available mapping to an individual student's personal information. However, if a school generates identification numbers based on student social security numbers, then the identification numbers should be considered private.

Security and privacy concerns often seem to compete with each other. From a security perspective, you should record all important operations so that any anomalous activity can later be identified. However, when private data is involved, this practice can create risk.

Although there are many ways in which private data can be handled unsafely, a common risk stems from misplaced trust. Programmers often trust the operating environment in which a program runs, and therefore believe that it is acceptable to store private information on the file system, in the registry, or in other locally-controlled resources. However, even if access to certain resources is restricted, this does not guarantee that the individuals who do have access can be trusted. For example, in 2004, an unscrupulous employee at AOL sold approximately 92 million private customer email addresses to a spammer marketing an offshore gambling web site [1].

In response to such high-profile exploits, the collection and management of private data is becoming increasingly regulated. Depending on its location, the type of business it conducts, and the nature of any private data it handles, an organization may be required to comply with one or more of the following federal and state regulations:

- Safe Harbor Privacy Framework [3]

- Gramm-Leach Bliley Act (GLBA) [4]

Copyright 2021 Micro Focus or one of its affiliates.

- Health Insurance Portability and Accountability Act (HIPAA) [5]

- California SB-1386 [6]

Despite these regulations, privacy violations continue to occur with alarming frequency.

Recommendations:

When security and privacy demands clash, privacy should usually be given the higher priority. To accomplish this and still maintain required security information, cleanse any private information before it exits the program.

To enforce good privacy management, develop and strictly adhere to internal privacy guidelines. The guidelines should specifically describe how an application should handle private data. If your organization is regulated by federal or state law, ensure that your privacy guidelines are sufficiently strenuous to meet the legal requirements. Even if your organization is not regulated, you must protect private information or risk losing customer confidence.

The best policy with respect to private data is to minimize its exposure. Applications, processes, and employees should not be granted access to any private data unless the access is required for the tasks that they are to perform. Just as the principle of least privilege dictates that no operation should be performed with more than the necessary privileges, access to private data should be restricted to the smallest possible group.

Tips:

1. As part of any thorough audit for privacy violations, ensure that custom rules are written to identify all sources of private or otherwise sensitive information entering the program. Most sources of private data cannot be identified automatically. Without custom rules, your check for privacy violations is likely to be substantially incomplete.

2. Due to the dynamic nature of PHP, you may see a large number of findings in PHP library files. Consider using a filter file to hide specific findings from view. For instructions on creating a filter file, see Advanced Options in the Fortify Static Code Analyzer User Guide.

class.pop3.php, line 334 (Privacy Violation)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Security Features		
Abstract:	The file class.pop3.php mishandles confidential information on line 334, which can compromise user privacy and is often illegal.		

Source: class.pop3.php:188 Read $this->password()

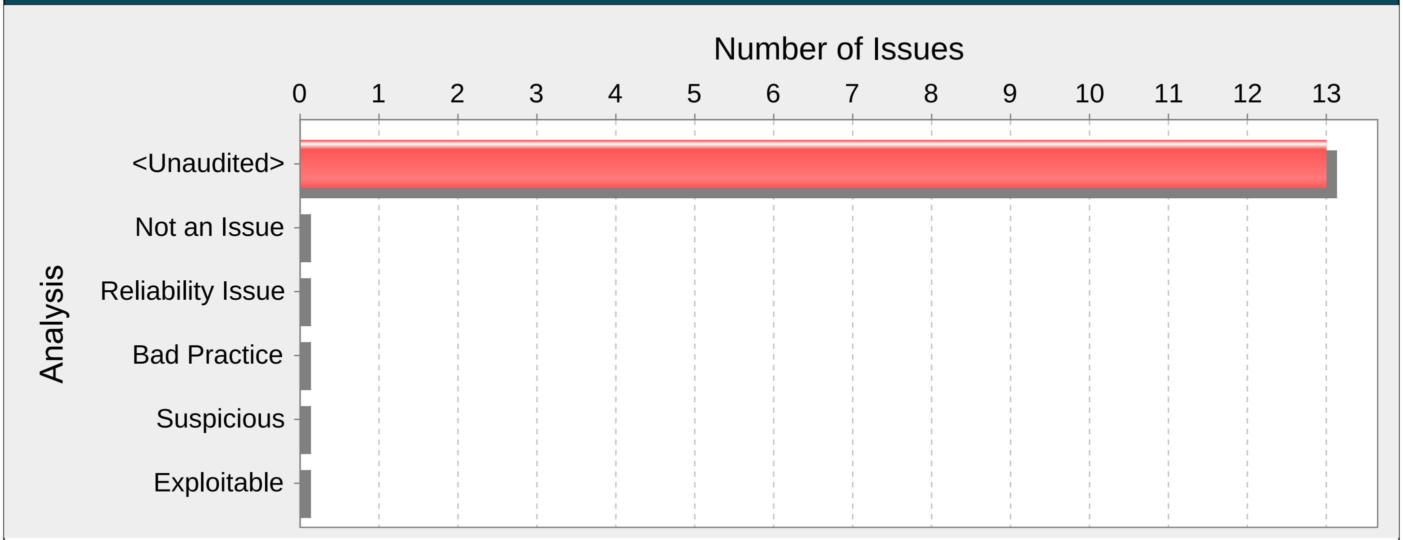
```
186            $result = $this->connect($this->host, $this->port, $this->tval);
187            if ($result) {
188                $login_result = $this->login($this->username, $this->password);
189                if ($login_result) {
190                    $this->disconnect();
```

Sink: class.pop3.php:334 builtin_echo()

```
332            if ($this->pop_conn) {
333                if ($this->do_debug >= 2) { //Show client messages when debug >= 2
334                    echo "Client -> Server: $string";
335                }
336                return fwrite($this->pop_conn, $string, strlen($string));
```

Copyright 2021 Micro Focus or one of its affiliates.

Category: Password Management: Empty Password (13 Issues)



Abstract:

Empty passwords may compromise system security in a way that cannot be easily remedied.

Explanation:

It is never a good idea to assign an empty string to a password variable. If the empty password is used to successfully authenticate against another system, then the corresponding account's security is likely compromised because it accepts an empty password. If the empty password is merely a placeholder until a legitimate value can be assigned to the variable, then it can confuse anyone unfamiliar with the code and potentially cause problems on unexpected control flow paths.

Example: The following code attempts to connect to a database with an empty password.

<?php

...

$connection = mysql_connect($host, 'scott', '');

...

?>

If the code in the Example succeeds, it indicates that the database user account "scott" is configured with an empty password, which an attacker can easily guess. After the program ships, updating the account to use a non-empty password will require a code change.

Recommendations:

Always read stored password values from encrypted, external resources and assign password variables meaningful values. Ensure that sensitive resources are never protected with empty or null passwords.

Starting with Microsoft(R) Windows(R) 2000, Microsoft(R) provides Windows Data Protection Application Programming Interface (DPAPI), which is an OS-level service that protects sensitive application data, such as passwords and private keys [1].

Tips:

1. Avoid empty passwords in source code and avoid using default passwords.

2. To identify null, empty, or hardcoded passwords, default rules only consider fields and variables that contain the word password. However, the Fortify Custom Rules Editor provides the Password Management wizard that makes it easy to create rules for detecting password management issues on custom-named fields and variables.

3. Due to the dynamic nature of PHP, you may see a large number of findings in PHP library files. Consider using a filter file to hide specific findings from view. For instructions on creating a filter file, see Advanced Options in the Fortify Static Code Analyzer User Guide.

redit.php, line 60 (Password Management: Empty Password)

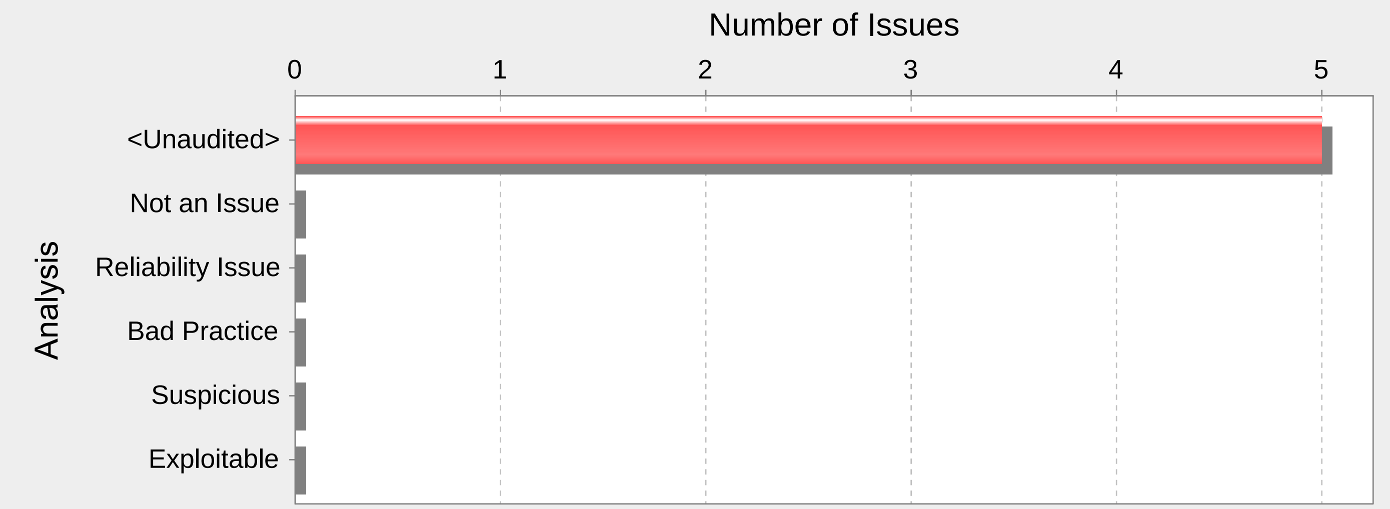
Fortify Priority:	High	**Folder**	High
Kingdom:	Security Features		
Abstract:	Empty passwords may compromise system security in a way that cannot be easily remedied.		
Sink:	redit.php:60 mysql_connect()		

```
58
59          $names =  (isset($_REQUEST['searchText']) ?  $_REQUEST['searchText'] : '');
60         @mysql_connect('localhost','root','');
```

```
61                        mysql_select_db('host_29723_mirbase') or die('db not selected');
```

Copyright 2021 Micro Focus or one of its affiliates.

Category: Cross-Site Scripting: DOM (5 Issues)



Abstract:

The method wd() in f.js sends unvalidated data to a web browser on line 1, which can result in the browser executing malicious code.

Explanation:

Cross-site scripting (XSS) vulnerabilities occur when:

1. Data enters a web application through an untrusted source. In the case of DOM-based XSS, data is read from a URL parameter or other value within the browser and written back into the page with client-side code. In the case of reflected XSS, the untrusted source is typically a web request, while in the case of persisted (also known as stored) XSS it is typically a database or other back-end data store.

2. The data is included in dynamic content that is sent to a web user without validation. In the case of DOM-based XSS, malicious content is executed as part of DOM (Document Object Model) creation, whenever the victim's browser parses the HTML page.

The malicious content sent to the web browser often takes the form of a JavaScript segment, but can also include HTML, Flash or any other type of code that the browser executes. The variety of attacks based on XSS is almost limitless, but they commonly include transmitting private data such as cookies or other session information to the attacker, redirecting the victim to web content controlled by the attacker, or performing other malicious operations on the user's machine under the guise of the vulnerable site.

Example 1: The following JavaScript code segment reads an employee ID, eid, from a URL and displays it to the user.

```
<SCRIPT>
var pos=document.URL.indexOf("eid=")+4;
document.write(document.URL.substring(pos,document.URL.length));
</SCRIPT>
```

Example 2: Consider the HTML form:

```
<div id="myDiv">
Employee ID: <input type="text" id="eid"><br>
...
<button>Show results</button>
</div>
<div id="resultsDiv">
...
</div>
```

The following jQuery code segment reads an employee ID from the form, and displays it to the user.

```
$(document).ready(function(){
$("#myDiv").on("click", "button", function(){
var eid = $("#eid").val();
$("resultsDiv").append(eid);
...
```

Copyright 2021 Micro Focus or one of its affiliates.

```
});
});
```

These code examples operate correctly if the employee ID from the text input with ID eid contains only standard alphanumeric text. If eid has a value that includes metacharacters or source code, then the code will be executed by the web browser as it displays the HTTP response.

Example 3: The following code shows an example of a DOM-based XSS within a React application:

```
let element = JSON.parse(getUntrustedInput());
ReactDOM.render(<App>
{element}
</App>);
```

In Example 3, if an attacker can control the entire JSON object retrieved from getUntrustedInput(), they may be able to make React render element as a component, and therefore can pass an object with dangerouslySetInnerHTML with their own controlled value, a typical cross-site scripting attack.

Initially these might not appear to be much of a vulnerability. After all, why would someone provide input containing malicious code to run on their own computer? The real danger is that an attacker will create the malicious URL, then use email or social engineering tricks to lure victims into visiting a link to the URL. When victims click the link, they unwittingly reflect the malicious content through the vulnerable web application back to their own computers. This mechanism of exploiting vulnerable web applications is known as Reflected XSS.

As the example demonstrates, XSS vulnerabilities are caused by code that includes unvalidated data in an HTTP response. There are three vectors by which an XSS attack can reach a victim:

- Data is read directly from the HTTP request and reflected back in the HTTP response. Reflected XSS exploits occur when an attacker causes a user to supply dangerous content to a vulnerable web application, which is then reflected back to the user and executed by the web browser. The most common mechanism for delivering malicious content is to include it as a parameter in a URL that is posted publicly or emailed directly to victims. URLs constructed in this manner constitute the core of many phishing schemes, whereby an attacker convinces victims to visit a URL that refers to a vulnerable site. After the site reflects the attacker's content back to the user, the content is executed and proceeds to transfer private information, such as cookies that might include session information, from the user's machine to the attacker or perform other nefarious activities.

- The application stores dangerous data in a database or other trusted data store. The dangerous data is subsequently read back into the application and included in dynamic content. Persistent XSS exploits occur when an attacker injects dangerous content into a data store that is later read and included in dynamic content. From an attacker's perspective, the optimal place to inject malicious content is in an area that is displayed to either many users or particularly interesting users. Interesting users typically have elevated privileges in the application or interact with sensitive data that is valuable to the attacker. If one of these users executes malicious content, the attacker may be able to perform privileged operations on behalf of the user or gain access to sensitive data belonging to the user.

- A source outside the application stores dangerous data in a database or other data store, and the dangerous data is subsequently read back into the application as trusted data and included in dynamic content.

Recommendations:

The solution to prevent XSS is to ensure that validation occurs in the required places and that relevant properties are set to prevent vulnerabilities.

Because XSS vulnerabilities occur when an application includes malicious data in its output, one logical approach is to validate data immediately before it leaves the application. However, because web applications often have complex and intricate code for generating dynamic content, this method is prone to errors of omission (missing validation). An effective way to mitigate this risk is to also perform input validation for XSS.

Web applications must validate all input to prevent other vulnerabilities, such as SQL injection, so augmenting an application's existing input validation mechanism to include checks for XSS is generally relatively easy. Despite its value, input validation for XSS does not take the place of rigorous output validation. An application might accept input through a shared data store or other trusted source, and that data store might accept input from a source that does not perform adequate input validation. Therefore, the application cannot implicitly rely on the safety of this or any other data. This means that the best way to prevent XSS vulnerabilities is to validate everything that enters the application and leaves the application destined for the user.

The most secure approach to validation for XSS is to create an allow list of safe characters that can appear in HTTP content and accept input composed exclusively of characters in the approved set. For example, a valid username might only include alphanumeric characters or a phone number might only include digits 0-9. However, this solution is often infeasible in web applications because many characters that have special meaning to the browser must be considered valid input after they are encoded, such as a web design bulletin board that must accept HTML fragments from its users.

A more flexible, but less secure approach is to implement a deny list, which selectively rejects or escapes potentially dangerous characters before using the input. To form such a list, you first need to understand the set of characters that hold special meaning for web browsers. Although the HTML standard defines which characters have special meaning, many web browsers try to correct common mistakes in HTML and might treat other characters as special in certain contexts. This is why we do not recommend the use of deny lists as a means to prevent XSS. The CERT(R) Coordination Center at the Software Engineering Institute at Carnegie Mellon University provides the following details about special characters in various contexts [1]:

In the content of a block-level element (in the middle of a paragraph of text):

Copyright 2021 Micro Focus or one of its affiliates.

- "<" is special because it introduces a tag.

- "&" is special because it introduces a character entity.

- ">" is special because some browsers treat it as special, on the assumption that the author of the page intended to include an opening "<", but omitted it in error.

The following principles apply to attribute values:

- In attribute values enclosed in double quotes, the double quotes are special because they mark the end of the attribute value.

- In attribute values enclosed in single quote, the single quotes are special because they mark the end of the attribute value.

- In attribute values without any quotes, white-space characters, such as space and tab, are special.

- "&" is special when used with certain attributes, because it introduces a character entity.

In URLs, for example, a search engine might provide a link within the results page that the user can click to re-run the search. This can be implemented by encoding the search query inside the URL, which introduces additional special characters:

- Space, tab, and new line are special because they mark the end of the URL.

- "&" is special because it either introduces a character entity or separates CGI parameters.

- Non-ASCII characters (that is, everything greater than 127 in the ISO-8859-1 encoding) are not allowed in URLs, so they are considered to be special in this context.

- The "%" symbol must be filtered from input anywhere parameters encoded with HTTP escape sequences are decoded by server-side code. For example, "%" must be filtered if input such as "%68%65%6C%6C%6F" becomes "hello" when it appears on the web page.

Within the body of a <SCRIPT> </SCRIPT>:

- Semicolons, parentheses, curly braces, and new line characters must be filtered out in situations where text could be inserted directly into a pre-existing script tag.

Server-side scripts:

- Server-side scripts that convert any exclamation characters (!) in input to double-quote characters (") on output might require additional filtering.

Other possibilities:

- If an attacker submits a request in UTF-7, the special character '<' appears as '+ADw-' and might bypass filtering. If the output is included in a page that does not explicitly specify an encoding format, then some browsers try to intelligently identify the encoding based on the content (in this case, UTF-7).

After you identify the correct points in an application to perform validation for XSS attacks and what special characters the validation should consider, the next challenge is to identify how your validation handles special characters. If special characters are not considered valid input to the application, then you can reject any input that contains special characters as invalid. A second option is to remove special characters with filtering. However, filtering has the side effect of changing any visual representation of the filtered content and might be unacceptable in circumstances where the integrity of the input must be preserved for display.

If input containing special characters must be accepted and displayed accurately, validation must encode any special characters to remove their significance. A complete list of ISO 8859-1 encoded values for special characters is provided as part of the official HTML specification [2].

Many application servers attempt to limit an application's exposure to cross-site scripting vulnerabilities by providing implementations for the functions responsible for setting certain specific HTTP response content that perform validation for the characters essential to a cross-site scripting attack. Do not rely on the server running your application to make it secure. For any developed application, there are no guarantees about which application servers it will run on during its lifetime. As standards and known exploits evolve, there are no guarantees that application servers will continue to stay in sync.

Tips:

1. The Fortify Secure Coding Rulepacks warn about SQL Injection and Access Control: Database issues when untrusted data is written to a database and also treat the database as a source of untrusted data, which can lead to XSS vulnerabilities. If the database is a trusted resource in your environment, use custom filters to filter out dataflow issues that include the DATABASE taint flag or originate from database sources. Nonetheless, it is often still a good idea to validate everything read from the database.

2. Even though URL encoding untrusted data protects against many XSS attacks, some browsers (specifically, Internet Explorer 6 and 7 and possibly others) automatically decode content at certain locations within the Document Object Model (DOM) prior to passing it to the JavaScript interpreter. To reflect this danger, the Rulepacks no longer treat URL encoding routines as sufficient to protect against cross-site scripting. Data values that are URL encoded and subsequently output will cause Fortify to report Cross-Site Scripting: Poor Validation vulnerabilities.

Copyright 2021 Micro Focus or one of its affiliates.

3. Older versions of React are more susceptible to cross-site scripting attacks by controlling an entire component. Newer versions use Symbols to identify a React component, which prevents the exploit, however older browsers that do not have Symbol support (natively, or through polyfills), such as all versions of Internet Explorer, are still vulnerable. Other types of cross-site scripting attacks are valid for all browsers and versions of React.

f.js, line 1 (Cross-Site Scripting: DOM)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Input Validation and Representation		

Abstract: The method wd() in f.js sends unvalidated data to a web browser on line 1, which can result in the browser executing malicious code.

Source: f.js:1 getItem()

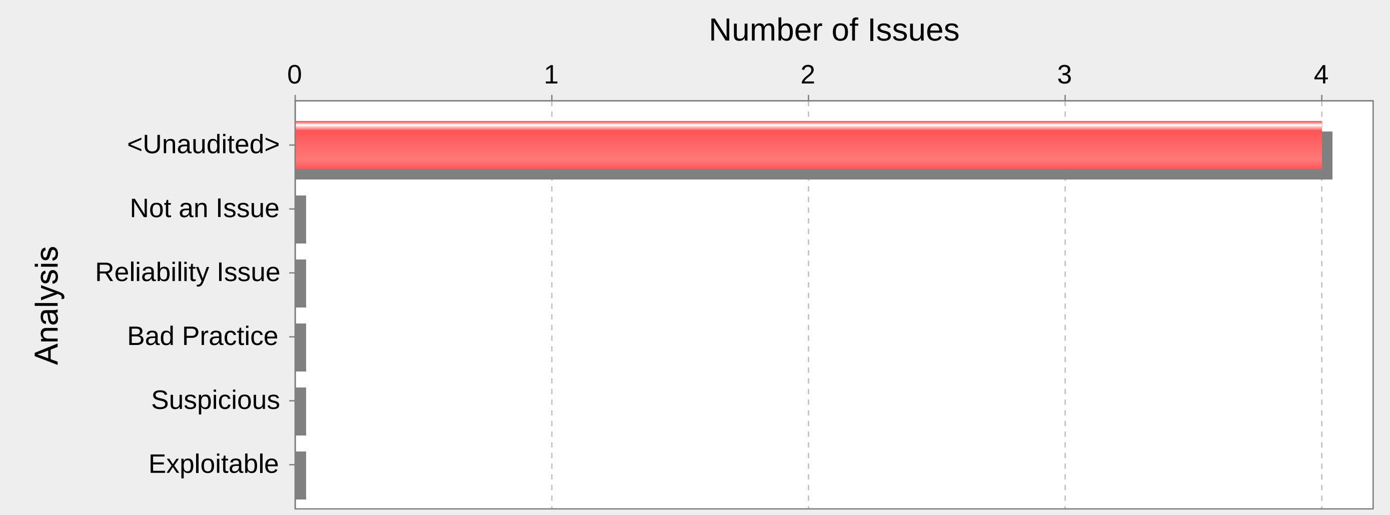
```
1    (function(){var l,aa="function"==typeof
Object.create?Object.create:function(a){function b(){}b.prototype=a;return new
b},ba;if("function"==typeof Object.setPrototypeOf)ba=Object.setPrototypeOf;else{var
ca;a:{var da={na:!0},ea={};try{ea.__proto__=da;ca=ea.na;break
a}catch(a){}ca=!1}ba=ca?function(a,b){a.__proto__=b;if(a.__proto__!==b)throw new
TypeError(a+" is not extensible");return a}:null}var fa=ba;function
ha(a,b){a.prototype=aa(b.prototype);a.prototype.constructor=a;if(fa)fa(a,b);else
for(v...
```

Sink: f.js:1 write()

```
1    (function(){var l,aa="function"==typeof
Object.create?Object.create:function(a){function b(){}b.prototype=a;return new
b},ba;if("function"==typeof Object.setPrototypeOf)ba=Object.setPrototypeOf;else{var
ca;a:{var da={na:!0},ea={};try{ea.__proto__=da;ca=ea.na;break
a}catch(a){}ca=!1}ba=ca?function(a,b){a.__proto__=b;if(a.__proto__!==b)throw new
TypeError(a+" is not extensible");return a}:null}var fa=ba;function
ha(a,b){a.prototype=aa(b.prototype);a.prototype.constructor=a;if(fa)fa(a,b);else
for(v...
```

Copyright 2021 Micro Focus or one of its affiliates.

Category: Open Redirect (4 Issues)



Abstract:

The file f.js passes unvalidated data to an HTTP redirect function on line 1. Allowing unvalidated input to control the URL used in a redirect can aid phishing attacks.

Explanation:

Redirects allow web applications to direct users to different pages within the same application or to external sites. Applications utilize redirects to aid in site navigation and, in some cases, to track how users exit the site. Open redirect vulnerabilities occur when a web application redirects clients to any arbitrary URL that can be controlled by an attacker.

Attackers might utilize open redirects to trick users into visiting a URL to a trusted site, but then redirecting them to a malicious site. By encoding the URL, an attacker can make it difficult for end-users to notice the malicious destination of the redirect, even when it is passed as a URL parameter to the trusted site. Open redirects are often abused as part of phishing scams to harvest sensitive end-user data.

Example 1: The following JavaScript code instructs the user's browser to open a URL read from the dest request parameter when a user clicks the link.

```
...
strDest = form.dest.value;
window.open(strDest,"myresults");
...
```

If a victim received an email instructing them to follow a link to "http://trusted.example.com/ecommerce/redirect.asp?dest=www.wilyhacker.com", the user would likely click on the link believing they would be transferred to the trusted site. However, when the victim clicks the link, the code in Example 1 will redirect the browser to "http://www.wilyhacker.com".

Many users have been educated to always inspect URLs they receive in emails to make sure the link specifies a trusted site they know. However, if the attacker Hex encoded the destination url as follows:

"http://trusted.example.com/ecommerce/redirect.asp?dest=%77%69%6C%79%68%61%63%6B%65%72%2E%63%6F%6D"

then even a savvy end-user may be fooled into following the link.

Recommendations:

Unvalidated user input should not be allowed to control the destination URL in a redirect. Instead, use a level of indirection: create a list of legitimate URLs that users are allowed to specify, and only allow users to select from the list. With this approach, input provided by users is never used directly to specify a URL for redirects.

Example 2: The following code references an array populated with valid URLs. The link the user clicks passes in the array index that corresponds to the desired URL.

```
...
strDest = form.dest.value;
if((strDest.value != null)||(strDest.value.length!=0))
{
if((strDest >= 0) && (strDest <= strURLArray.length -1 ))
{
strFinalURL = strURLArray[strDest];
window.open(strFinalURL,"myresults");
```

Copyright 2021 Micro Focus or one of its affiliates.

```
}
}
...
```

In some situations this approach is impractical because the set of legitimate URLs is too large or too hard to keep track of. In such cases, use a similar approach to restrict the domains that users can be redirected to, which can at least prevent attackers from sending users to malicious external sites.

f.js, line 1 (Open Redirect)

Fortify Priority:	High	**Folder**	High
Kingdom:	Input Validation and Representation		

Abstract: The file f.js passes unvalidated data to an HTTP redirect function on line 1. Allowing unvalidated input to control the URL used in a redirect can aid phishing attacks.

Source: f.js:1 getItem()

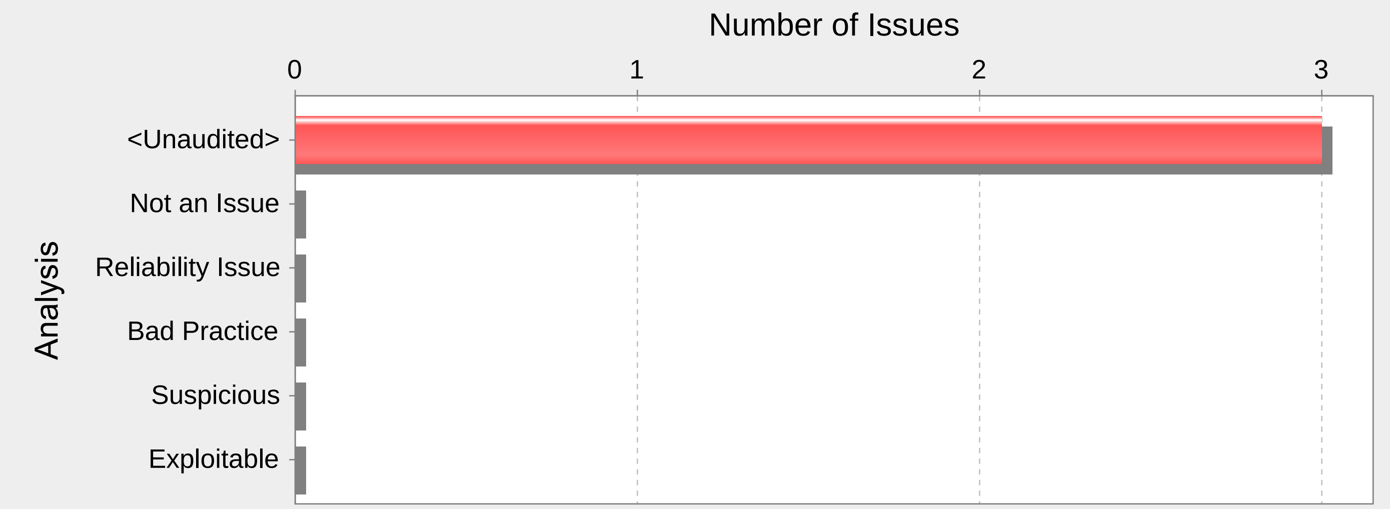
```
1    (function(){var l,aa="function"==typeof
     Object.create?Object.create:function(a){function b(){}b.prototype=a;return new
     b},ba;if("function"==typeof Object.setPrototypeOf)ba=Object.setPrototypeOf;else{var
     ca;a:{var da={na:!0},ea={};try{ea.__proto__=da;ca=ea.na;break
     a}catch(a){}ca=!1}ba=ca?function(a,b){a.__proto__=b;if(a.__proto__!==b)throw new
     TypeError(a+" is not extensible");return a}:null}var fa=ba;function
     ha(a,b){a.prototype=aa(b.prototype);a.prototype.constructor=a;if(fa)fa(a,b);else
     for(v...
```

Sink: f.js:1 Assignment to a.src()

```
1    (function(){var l,aa="function"==typeof
     Object.create?Object.create:function(a){function b(){}b.prototype=a;return new
     b},ba;if("function"==typeof Object.setPrototypeOf)ba=Object.setPrototypeOf;else{var
     ca;a:{var da={na:!0},ea={};try{ea.__proto__=da;ca=ea.na;break
     a}catch(a){}ca=!1}ba=ca?function(a,b){a.__proto__=b;if(a.__proto__!==b)throw new
     TypeError(a+" is not extensible");return a}:null}var fa=ba;function
     ha(a,b){a.prototype=aa(b.prototype);a.prototype.constructor=a;if(fa)fa(a,b);else
     for(v...
```

Copyright 2021 Micro Focus or one of its affiliates.

Category: Path Manipulation (3 Issues)



Abstract:

Attackers can control the file system path argument to copy() at price.php line 19, which allows them to access or modify otherwise protected files.

Explanation:

Path manipulation errors occur when the following two conditions are met:

1. An attacker can specify a path used in an operation on the file system.

2. By specifying the resource, the attacker gains a capability that would not otherwise be permitted.

For example, the program might give the attacker the ability to overwrite the specified file or run with a configuration controlled by the attacker.

Example 1: The following code uses input from an HTTP request to create a file name. The programmer has not considered the possibility that an attacker could provide a file name such as "../../tomcat/conf/server.xml", which causes the application to delete one of its own configuration files.

```
$rName = $_GET['reportName'];
$rFile = fopen("/usr/local/apfr/reports/" . rName,"a+");
...
unlink($rFile);
```

Example 2: The following code uses input from a configuration file to determine which file to open and echo back to the user. If the program runs with adequate privileges and malicious users can change the configuration file, they can use the program to read any file on the system that ends with the extension .txt.

```
...
$filename = $CONFIG_TXT['sub'] . ".txt";
$handle = fopen($filename,"r");
$amt = fread($handle, filesize($filename));
echo $amt;
...
```

Recommendations:

The best way to prevent path manipulation is with a level of indirection: create a list of legitimate values from which the user must select. With this approach, the user-provided input is never used directly to specify the resource name.

In some situations this approach is impractical because the set of legitimate resource names is too large or too hard to maintain. Programmers often resort to implementing a deny list in these situations. A deny list is used to selectively reject or escape potentially dangerous characters before using the input. However, any such list of unsafe characters is likely to be incomplete and will almost certainly become out of date. A better approach is to create a list of characters that are permitted to appear in the resource name and accept input composed exclusively of characters in the approved set.

Tips:

1. If the program performs custom input validation to your satisfaction, use the Fortify Custom Rules Editor to create a cleanse rule for the validation routine.

Copyright 2021 Micro Focus or one of its affiliates.

2. Implementation of an effective deny list is notoriously difficult. One should be skeptical if validation logic requires implementing a deny list. Consider different types of input encoding and different sets of metacharacters that might have special meaning when interpreted by different operating systems, databases, or other resources. Determine whether or not the deny list can be updated easily, correctly, and completely if these requirements ever change.

3. Due to the dynamic nature of PHP, you may see a large number of findings in PHP library files. Consider using a filter file to hide specific findings from view. For instructions on creating a filter file, see Advanced Options in the Fortify Static Code Analyzer User Guide.

price.php, line 19 (Path Manipulation)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Input Validation and Representation		

Abstract: Attackers can control the file system path argument to copy() at price.php line 19, which allows them to access or modify otherwise protected files.

Source: price.php:19 Read $_FILES['filename']['name']()

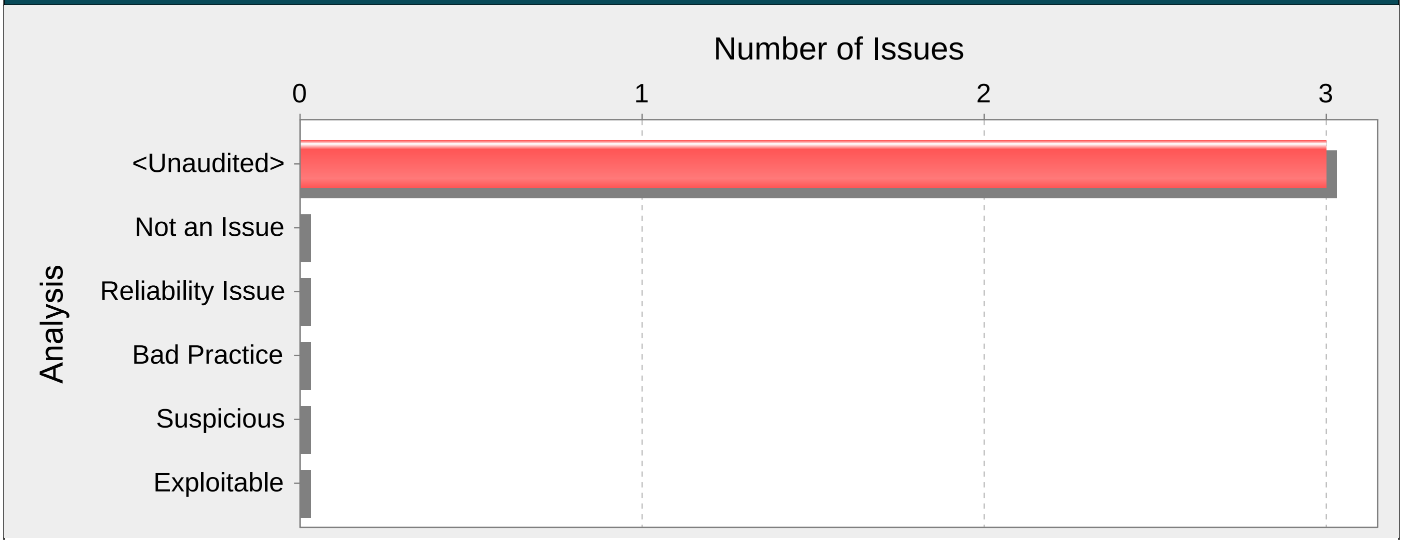
```
17          exit;
18        }
19        if(copy($_FILES["filename"]["tmp_name"],$path.$_FILES["filename"]["name"])){
20            echo(" "."<b>".$_FILES["filename"]["name"]."</b>"."  !<br>");
21        }   else{
```

Sink: price.php:19 copy()

```
17          exit;
18        }
19        if(copy($_FILES["filename"]["tmp_name"],$path.$_FILES["filename"]["name"])){
20            echo(" "."<b>".$_FILES["filename"]["name"]."</b>"."  !<br>");
21        }   else{
```

Copyright 2021 Micro Focus or one of its affiliates.

Category: Privacy Violation: Autocomplete (3 Issues)



Abstract:

The form in admin111.php uses autocompletion on line 64, which allows some browsers to retain sensitive information in their history.

Explanation:

With autocompletion enabled, some browsers retain user input across sessions, which could allow someone using the computer after the initial user to see information previously submitted.

Recommendations:

Explicitly disable autocompletion on forms or sensitive inputs. By disabling autocompletion, information previously entered will not be presented back to the user as they type. It will also disable the "remember my password" functionality of most major browsers.

Example 1: In an HTML form, disable autocompletion for all input fields by explicitly setting the value of the autocomplete attribute to off on the form tag.

```
<form method="post" autocomplete="off">
Address: <input name="address" />
Password: <input name="password" type="password" />
</form>
```

Example 2: Alternatively, disable autocompletion for specific input fields by explicitly setting the value of the autocomplete attribute to off on the corresponding tags.

```
<form method="post">
Address: <input name="address" />
Password: <input name="password" type="password" autocomplete="off"/>
</form>
```

Note that the default value of the autocomplete attributed is on. Therefore do not omit the attribute when dealing with sensitive inputs.

admin111.php, line 64 (Privacy Violation: Autocomplete)

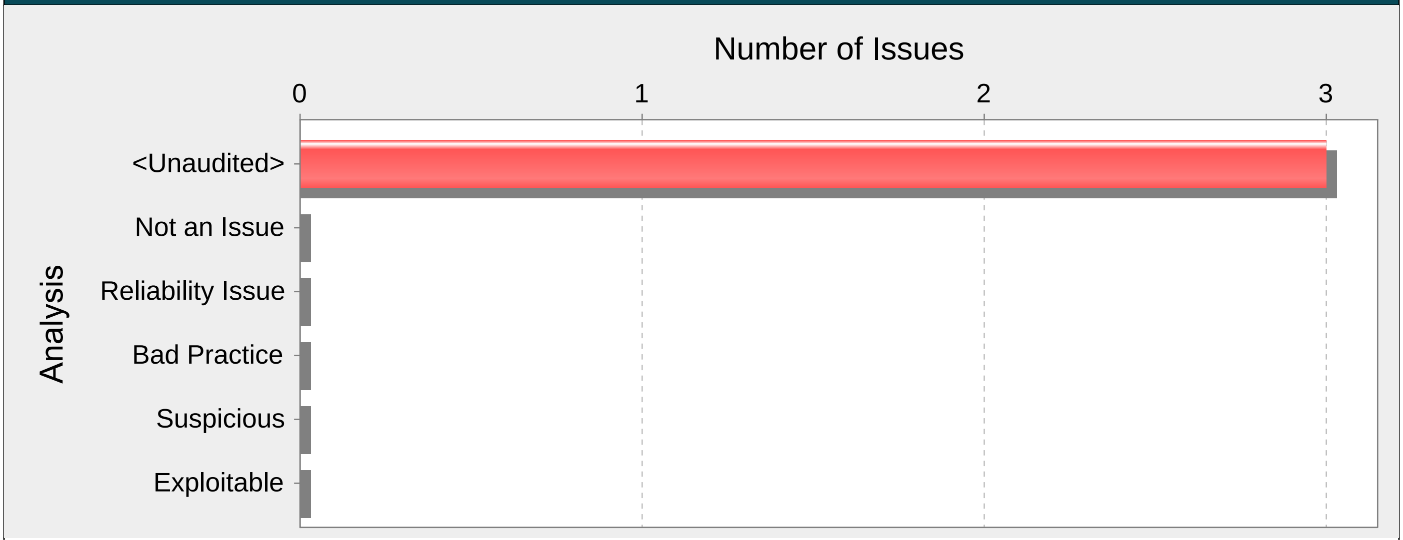
Fortify Priority:	High	**Folder**	High
Kingdom:	Security Features		

Abstract: The form in admin111.php uses autocompletion on line 64, which allows some browsers to retain sensitive information in their history.

Sink: admin111.php:64 null()

```
62
63          <input type="text" name="login" required placeholder=''/> <br>
64        <input type="password" name="password" required placeholder=''/> <br>
65  <br><a href="http://srv54/optimabank-dba.kg/admin.pdf" class="url-min" title="manual">
     </a><br><br>
66  <a href="index.php " class="url-min" title=" ">Back</a><br><br>
```

Category: System Information Leak: External (3 Issues)



Abstract:

The program might reveal system data or debugging information in example.php with a call to highlight_file() on line 164. The information revealed by highlight_file() could help an adversary form a plan of attack.

Explanation:

An external information leak occurs when system data or debugging information leaves the program to a remote machine via a socket or network connection.

Example 1: The following code writes an exception to the HTTP response:

```
<?php
...
echo "Server error! Printing the backtrace";
debug_print_backtrace();
...
?>
```

Depending upon the system configuration, this information can be dumped to a console, written to a log file, or exposed to a remote user. For example, with scripting mechanisms it is trivial to redirect output information from "Standard error" or "Standard output" into a file or another program. Alternatively, the system that the program runs on could have a remote logging mechanism such as a "syslog" server that sends the logs to a remote device. During development, you have no way of knowing where this information might end up being displayed.

In some cases, the error message provides the attacker with the precise type of attack to which the system is vulnerable. For example, a database error message can reveal that the application is vulnerable to a SQL injection attack. Other error messages can reveal more oblique clues about the system. In Example 1, the leaked information could imply information about the type of operating system, the applications installed on the system, and the amount of care that the administrators have put into configuring the program.

Recommendations:

Write error messages with security in mind. In production environments, turn off detailed error information in favor of brief messages. Restrict the generation and storage of detailed output that can help administrators and programmers diagnose problems. Debug traces can sometimes appear in non-obvious places (embedded in comments in the HTML for an error page, for example).

Even brief error messages that do not reveal stack traces or database dumps can potentially aid an attacker. For example, an "Access Denied" message can reveal that a file or user exists on the system.

Tips:

1. Do not rely on wrapper scripts, corporate IT policy, or quick-thinking system administrators to prevent system information leaks. Write software that is secure on its own.

2. This category of vulnerability does not apply to all types of programs. For example, if your application executes on a client machine where system information is already available to an attacker, or if you print system information only to a trusted log file, you can use Audit Guide to filter out this category from your scan results.

3. Due to the dynamic nature of PHP, you may see a large number of findings in PHP library files. Consider using a filter file to hide specific findings from view. For instructions on creating a filter file, see Advanced Options in the Fortify Static Code Analyzer User Guide.

example.php, line 164 (System Information Leak: External)

Copyright 2021 Micro Focus or one of its affiliates.

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Encapsulation		

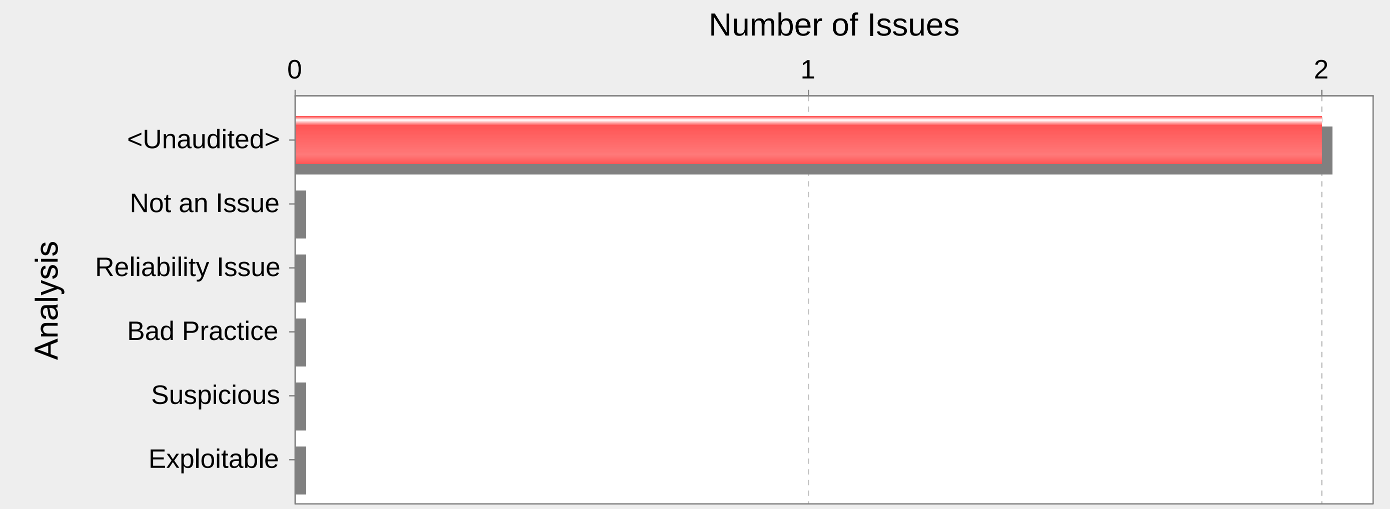
Abstract: The program might reveal system data or debugging information in example.php with a call to highlight_file() on line 164. The information revealed by highlight_file() could help an adversary form a plan of attack.

Sink: example.php:164 highlight_file()

```
162
163 <?php
164 highlight_file("../examples/MagicSquareExample.php");
165 include_once "includes/footer.php";
166 ?>
```

Copyright 2021 Micro Focus or one of its affiliates.

Category: Credential Management: Hardcoded API Credentials (2 Issues)



Abstract:

Hardcoded API credentials can compromise system security in a way that is not easy to remedy.

Explanation:

Never hardcode credentials, including usernames, passwords, API keys, API secrets, and API Tokens. Not only are hardcoded credentials visible to all of the project developers, they are extremely difficult to update. After the code is in production, the credentials cannot be changed without patching the software. If the credentials are compromised, the organization must choose between security and system availability.

Recommendations:

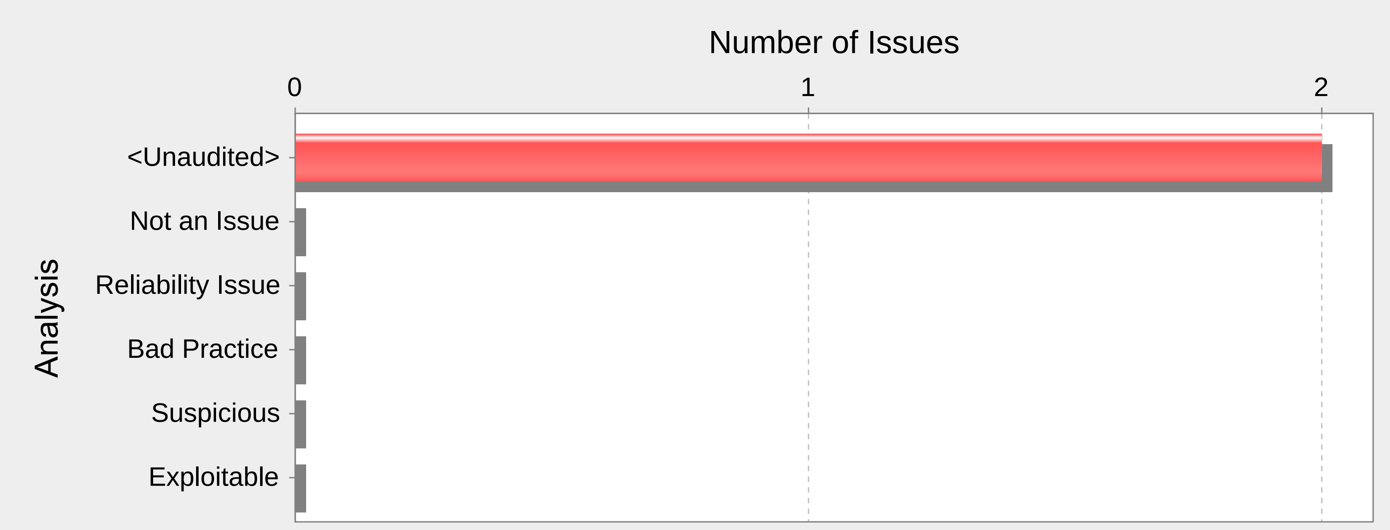
Make sure that API credentials are either loaded from a configuration file that is only available in the runtime environment or from environment variables.

analytics.js. , line 18 (Credential Management: Hardcoded API Credentials)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Security Features		
Abstract:	Hardcoded API credentials can compromise system security in a way that is not easy to remedy.		
Sink:	analytics.js.Без названия:18 null()		

Copyright 2021 Micro Focus or one of its affiliates.

Category: Dynamic Code Evaluation: Code Injection (2 Issues)



Abstract:

The file prototype-1.7.0.0.js interprets unvalidated user input as source code on line 1643. Interpreting user-controlled instructions at run-time can allow attackers to execute malicious code.

Explanation:

Many modern programming languages allow dynamic interpretation of source instructions. This capability allows programmers to perform dynamic instructions based on input received from the user. Code injection vulnerabilities occur when the programmer incorrectly assumes that instructions supplied directly from the user will perform only innocent operations, such as performing simple calculations on active user objects or otherwise modifying the user's state. However, without proper validation, a user might specify operations the programmer does not intend.

Example: In this classic code injection example, the application implements a basic calculator that allows the user to specify commands for execution.

```
...
userOp = form.operation.value;
calcResult = eval(userOp);
...
```

The program behaves correctly when the operation parameter is a benign value, such as "8 + 7 * 2", in which case the calcResult variable is assigned a value of 22. However, if an attacker specifies languages operations that are both valid and malicious, those operations would be executed with the full privilege of the parent process. Such attacks are even more dangerous when the underlying language provides access to system resources or allows execution of system commands. In the case of JavaScript, the attacker may utilize this vulnerability to perform a cross-site scripting attack.

Recommendations:

Avoid dynamic code interpretation whenever possible. If your program's functionality requires code to be interpreted dynamically, the likelihood of attack can be minimized by constraining the code your program will execute dynamically as much as possible, limiting it to an application- and context-specific subset of the base programming language.

If dynamic code execution is required, unvalidated user input should never be directly executed and interpreted by the application. Instead, use a level of indirection: create a list of legitimate operations and data objects that users are allowed to specify, and only allow users to select from the list. With this approach, input provided by users is never executed directly.

prototype-1.7.0.0.js, line 1643 (Dynamic Code Evaluation: Code Injection)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Input Validation and Representation		
Abstract:	The file prototype-1.7.0.0.js interprets unvalidated user input as source code on line 1643. Interpreting user-controlled instructions at run-time can allow attackers to execute malicious code.		
Source:	prototype-1.7.0.0.js:1643 Read responseText()		

```
1641      evalResponse: function() {
1642        try {
1643          return eval((this.transport.responseText || '').unfilterJSON());
1644        } catch (e) {
1645          this.dispatchException(e);
```

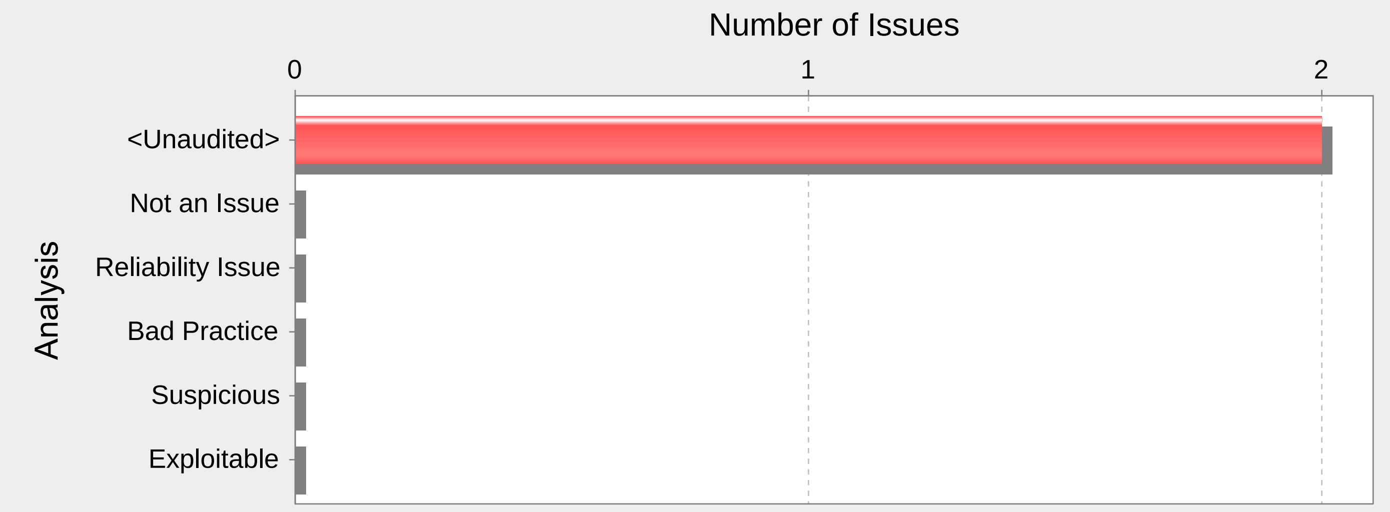
Sink: prototype-1.7.0.0.js:1643 eval()

Copyright 2021 Micro Focus or one of its affiliates.

```
                        evalResponse: function() {
                          try {
                            return eval((this.transport.responseText || '').unfilterJSON());
                          } catch (e) {
                            this.dispatchException(e);
```

Copyright 2021 Micro Focus or one of its affiliates.

Category: Dynamic Code Evaluation: Insecure Transport (2 Issues)



Abstract:

The file index.html on line 64 loads a script over an unencrypted channel.

Explanation:

Including executable content from a website over an unencrypted channel enables an attacker to perform a man-in-the-middle (MiTM) attack. This enables an attacker to load their own content that is executed as if it was part of the original website.

Example: Consider the following script tag:

<script src="http://www.example.com/js/fancyWidget.js"></script>

If an attacker is listening to the network traffic between the user and the server, the attacker can imitate or manipulate the content from www.example.com to load their own JavaScript.

Recommendations:

Control the code that your web pages load and if the code is coming from a separate domain, make sure the code is always loaded over a secure connection. Whenever possible, avoid including scripts or other artifacts from third-party sites.

index.html, line 64 (Dynamic Code Evaluation: Insecure Transport)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Security Features		

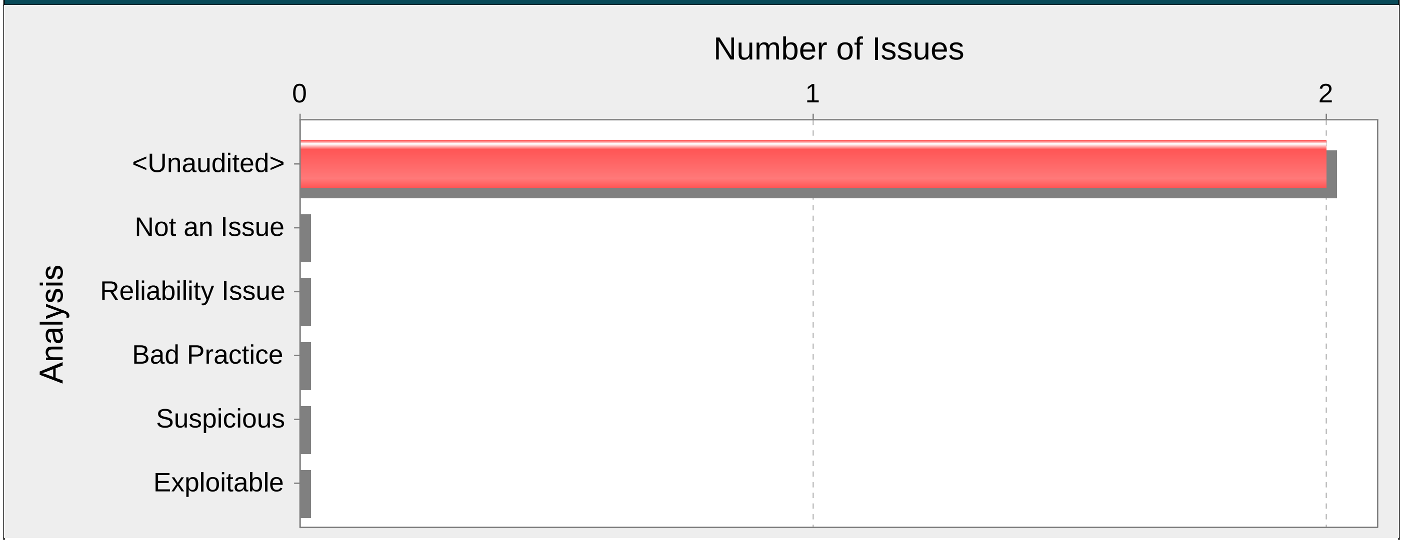
Abstract: The file index.html on line 64 loads a script over an unencrypted channel.

Sink: index.html:64 null()

```
62      <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js"></script>
63      <!-- IE10 viewport hack for Surface/desktop Windows 8 bug -->
64      <script src="http://getbootstrap.com/assets/js/ie10-viewport-bug-
workaround.js"></script>
65    </body>
66  </html>
```

Copyright 2021 Micro Focus or one of its affiliates.

Category: Key Management: Hardcoded Encryption Key (2 Issues)



Abstract:

Hardcoded encryption keys can compromise security in a way that is not easy to remedy.

Explanation:

Never hardcode an encryption key because it makes the encryption key visible to all of the project's developers, and makes fixing the problem extremely difficult. Changing the encryption key after the code is in production requires a software patch. If the account that the encryption key protects is compromised, the organization must choose between security and system availability.

Example 1: The following example shows an encryption key inside a .pem file:

```
...
-----BEGIN RSA PRIVATE KEY-----
MIICXwIBAAKBgQCtVacMo+w+TFOm0p8MlBWvwXtVRpF28V+o0RNPx5x/1TJTlKEl
...
DiJPJY2LNBQ7jS685mb6650JdvH8uQl6oeJ/aUmq63o2zOw=
-----END RSA PRIVATE KEY-----
...
```

Anyone with access to the code can see the encryption key. After the application has shipped, there is no way to change the encryption key unless the program is patched. An employee with access to this information can use it to break into the system. Any attacker with access to the application executable can extract the encryption key value.

Recommendations:

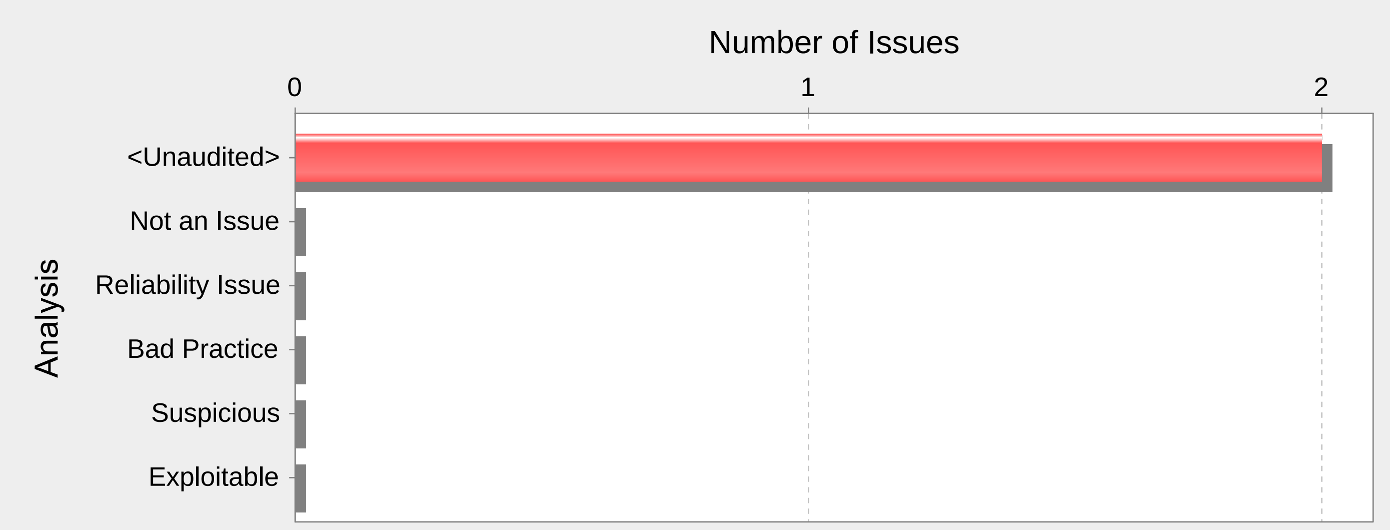
Never check in encryption keys to your source control system, and never hardcode them. Always obfuscate and manage encryption keys in an external source. Storing encryption keys in plain text anywhere on the system enables anyone with sufficient permissions to read and potentially misuse the encryption key.

tcpdf.crt, line 26 (Key Management: Hardcoded Encryption Key)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Security Features		
Abstract:	Hardcoded encryption keys can compromise security in a way that is not easy to remedy.		
Sink:	tcpdf.crt:26 null()		

Copyright 2021 Micro Focus or one of its affiliates.

Category: Privacy Violation: Shoulder Surfing (2 Issues)



Abstract:

The form in in.php writes the password in plain text to the screen on line 62.

Explanation:

A password need not be viewable to its owner, and must not be viewable to others. If a password is displayed in plain text, anyone in the vicinity could see and use it to compromise the system. In computer security, shoulder surfing refers to using direct observation techniques, such as looking over someone's shoulder, to get information. Shoulder surfing is particularly effective in crowded, public environments. This threat particularly applies to mobile devices, which are generally intended for use in all environments, both private and public.

Recommendations:

Never show a password in plain text. Obscure the characters of the field as dots or stars instead of legible characters.

Example: In HTML forms, set the type attribute to password for sensitive inputs.

<input name="password" type="password" />

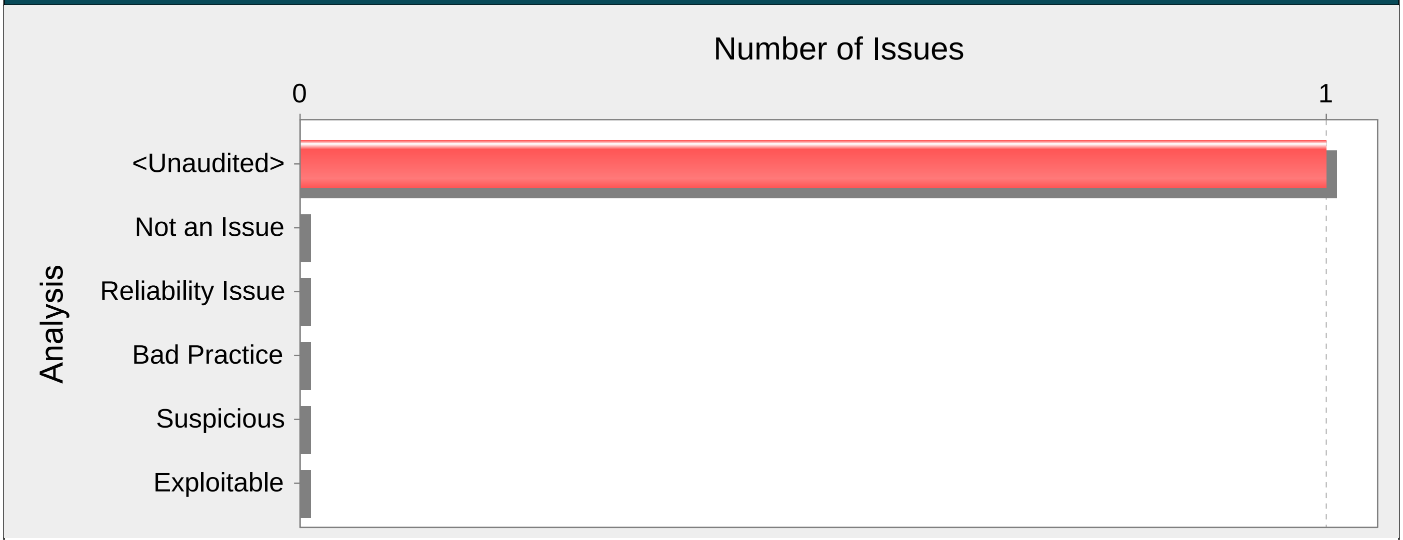
Note that the default value of the type attributed is text, not password. Thus do not omit the attribute when dealing with sensitive inputs.

in.php, line 62 (Privacy Violation: Shoulder Surfing)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Security Features		
Abstract:	The form in in.php writes the password in plain text to the screen on line 62.		
Sink:	in.php:62 null()		

```
60      <br>
61
62          <input type="text" name="password" required placeholder=''/> <br>
63         <input type="password" name="name" required placeholder=''/> <br>
```

Category: Dangerous Function (1 Issues)



Abstract:

The function mysql_escape_string() cannot be used safely. It should not be used.

Explanation:

Certain functions behave in dangerous ways regardless of how they are used. Functions in this category were often implemented without taking security concerns into account.

The mysql_escape_string() function is unsafe as it does not take into consideration the current character encoding set within the database. Thus, it may be vulnerable to multi-byte escaping issues. mysql_escape_string also does not escape the '%' and '_' characters.

Recommendations:

Functions that cannot be used safely should never be used. If any of these functions occur in new or legacy code, they must be removed and replaced with safe alternatives.

Tips:

1. Due to the dynamic nature of PHP, you may see a large number of findings in PHP library files. Consider using a filter file to hide specific findings from view. For instructions on creating a filter file, see Advanced Options in the Fortify Static Code Analyzer User Guide.

Database.php, line 300 (Dangerous Function)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	API Abuse		

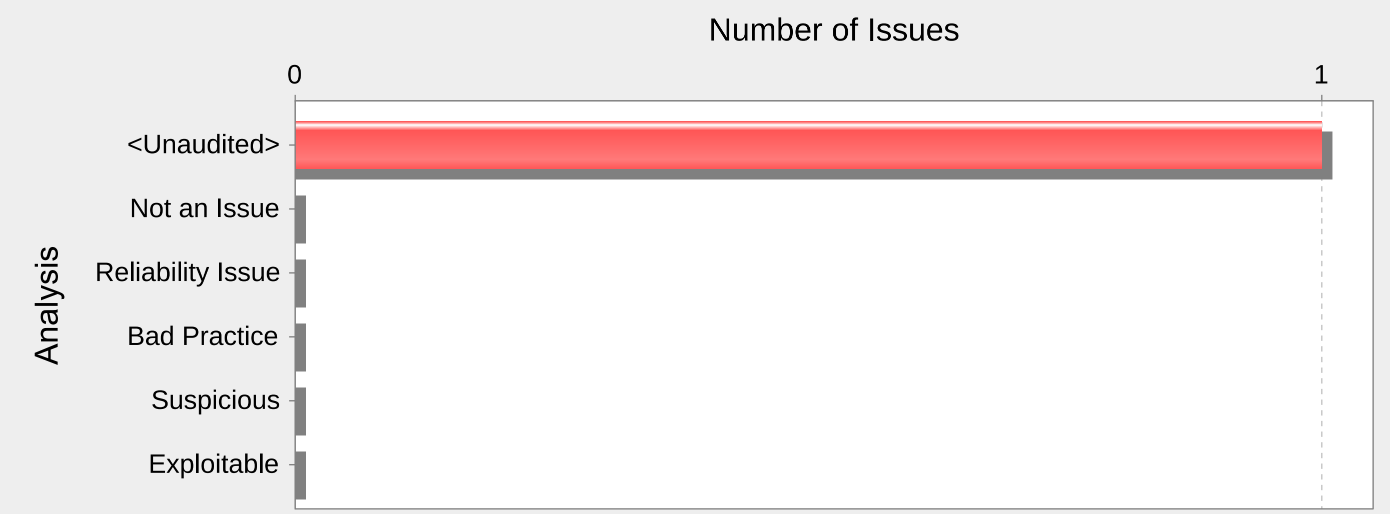
Abstract: The function mysql_escape_string() cannot be used safely. It should not be used.

Sink: Database.php:300 mysql_escape_string()

```
298        if (PHP_VERSION < 6) $theValue = get_magic_quotes_gpc() ? stripslashes($theValue) :
    $theValue;
299
300        $theValue = function_exists("mysql_real_escape_string") ?
    mysql_real_escape_string($theValue) : mysql_escape_string($theValue);
301
302        switch ($theType) {
```

Copyright 2021 Micro Focus or one of its affiliates.

Category: Key Management: Empty Encryption Key (1 Issues)



Abstract:

Empty encryption keys can compromise security in a way that cannot be easily remedied.

Explanation:

It is never a good idea to use an empty encryption key. Not only does using an empty encryption key significantly reduce the protection afforded by a good encryption algorithm, but it also makes fixing the problem extremely difficult. After the offending code is in production, the empty encryption key cannot be changed without patching the software. If an account protected by the empty encryption key is compromised, the owners of the system must choose between security and availability.

Example: The following code initializes an encryption key variable to an empty string.

```
...
$encryption_key = '';
$filter = new Zend_Filter_Encrypt($encryption_key);
$filter->setVector('myIV');
$encrypted = $filter->filter('text_to_be_encrypted');
print $encrypted;
...
```

Not only will anyone who has access to the code be able to determine that it uses an empty encryption key, but anyone with even basic cracking techniques is much more likely to successfully decrypt any encrypted data. After the program ships, a software patch is required to change the empty encryption key. An employee with access to this information can use it to break into the system. Even if attackers only had access to the application's executable, they could extract evidence of the use of an empty encryption key.

Recommendations:

Encryption keys should never be empty and should generally be obfuscated and managed in an external source. Storing encryption keys in plain text, empty or otherwise, anywhere on the system allows anyone with sufficient permissions to read and potentially misuse the encryption key.

Starting with Microsoft(R) Windows(R) 2000, Microsoft(R) provides Windows Data Protection Application Programming Interface (DPAPI), which is an OS-level service that protects sensitive application data, such as passwords and private keys [1].

Tips:

1. Avoid empty string encryption keys in source code.

2. When identifying null, empty, or hardcoded encryption keys, default rules only consider fields and variables that contain the word enc_key, encryption_key, passphrase, or pass_phrase. However, the Fortify Custom Rules Editor provides the Password Management wizard that makes it easy to create rules for detecting key management issues on custom-named fields and variables.

3. Due to the dynamic nature of PHP, you may see a large number of findings in PHP library files. Consider using a filter file to hide specific findings from view. For instructions on creating a filter file, see Advanced Options in the Fortify Static Code Analyzer User Guide.

tcpdf.php, line 1802 (Key Management: Empty Encryption Key)

Fortify Priority:	High	**Folder**	High
Kingdom:	Security Features		

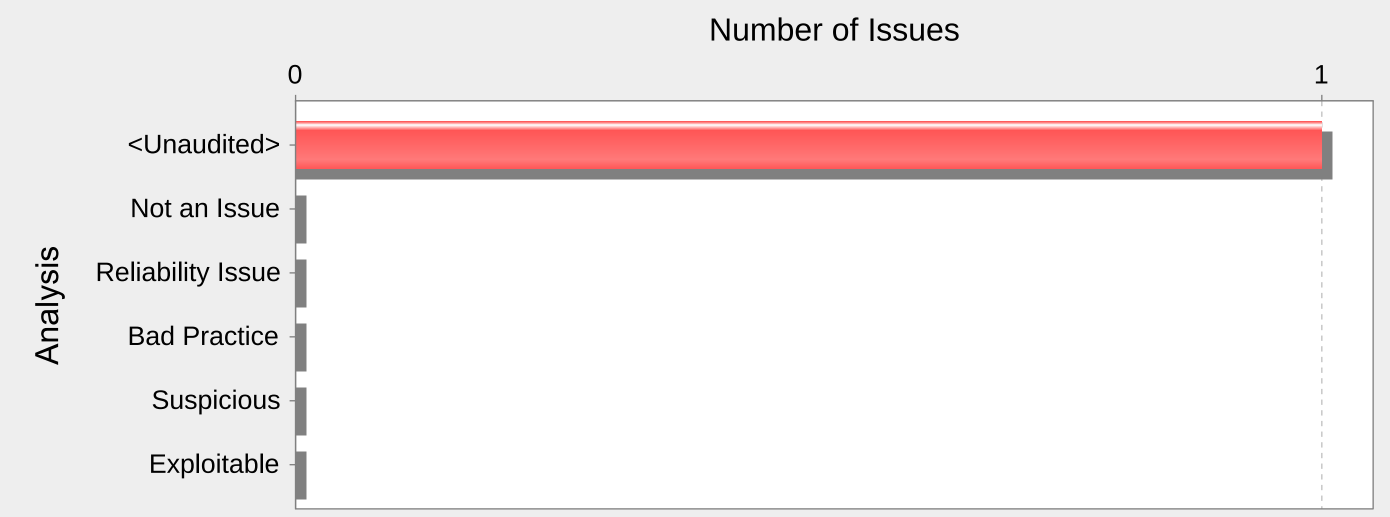
Copyright 2021 Micro Focus or one of its affiliates.

Abstract:	Empty encryption keys can compromise security in a way that cannot be easily remedied.
Sink:	tcpdf.php:1802 FieldAccess: last_enc_key()

```
1800        // encryption values
1801        $this->encrypted = false;
1802        $this->last_enc_key = '';
1803        // standard Unicode fonts
1804        $this->CoreFonts = array(
```

Copyright 2021 Micro Focus or one of its affiliates.

Category: Weak Encryption (1 Issues)



Abstract:

The call to mcrypt_decrypt() at tcpdf.php line 11638 uses a weak encryption algorithm that cannot guarantee the confidentiality of sensitive data.

Explanation:

Antiquated encryption algorithms such as DES no longer provide sufficient protection for use with sensitive data. Encryption algorithms rely on key size as one of the primary mechanisms to ensure cryptographic strength. Cryptographic strength is often measured by the time and computational power needed to generate a valid key. Advances in computing power have made it possible to obtain small encryption keys in a reasonable amount of time. For example, the 56-bit key used in DES posed a significant computational hurdle in the 1970s when the algorithm was first developed, but today DES can be cracked in less than a day using commonly available equipment.

Recommendations:

Use strong encryption algorithms with large key sizes to protect sensitive data. A strong alternative to DES is AES (Advanced Encryption Standard, formerly Rijndael). Before selecting an algorithm, first determine if your organization has standardized on a specific algorithm and implementation.

Tips:

1. Due to the dynamic nature of PHP, you may see a large number of findings in PHP library files. Consider using a filter file to hide specific findings from view. For instructions on creating a filter file, see Advanced Options in the Fortify Static Code Analyzer User Guide.

tcpdf.php, line 11638 (Weak Encryption)

Fortify Priority:	Critical	**Folder**	Critical
Kingdom:	Security Features		

Abstract: The call to mcrypt_decrypt() at tcpdf.php line 11638 uses a weak encryption algorithm that cannot guarantee the confidentiality of sensitive data.

Sink: tcpdf.php:11638 FunctionCall: mcrypt_decrypt()

```
11636    */
11637   protected function _RC4($key, $text) {
11638    if (function_exists('mcrypt_decrypt') AND ($out = @mcrypt_decrypt(MCRYPT_ARCFOUR,
        $key, $text, MCRYPT_MODE_STREAM, ''))) {
11639     // try to use mcrypt function if exist
11640     return $out;
```

Copyright 2021 Micro Focus or one of its affiliates.

Detailed Project Summary

Files Scanned

Code base location: F:/BD

Files Scanned:

1.php php 41 Lines 11.6 KB Oct 30, 2018, 11:40:50 PM

1/db_conn.php php 10 Lines Feb 22, 2017, 8:14:50 PM

1/form_file_load.php php 1 Lines Feb 22, 2017, 8:14:50 PM

1/index.php php 182 Lines 25.2 KB Mar 5, 2017, 3:39:32 AM

1/price.php php 20 Lines 2.2 KB Feb 27, 2017, 3:41:02 AM

1/reader.php php 326 Lines 47.9 KB Feb 12, 2017, 1:28:16 AM

123/123/FileSaver.js typescript 146 Lines 7.7 KB May 25, 2014, 9:11:44 PM

123/123/README.md generic   Apr 21, 2015, 7:43:24 PM

123/123/f.js typescript 1 Lines 59 KB Jan 15, 2019, 3:08:12 AM

123/123/f.txt generic 59 KB Jan 15, 2019, 3:06:31 AM

123/123/index.htm html 15 Lines 1.7 KB Jan 15, 2019, 4:20:05 AM

123/123/jquery.wordexport.js typescript 49 Lines 4 KB Mar 3, 2023, 4:05:02 AM

123/1234/FileSaver.js typescript 146 Lines 7.7 KB May 25, 2014, 9:11:44 PM

123/1234/README.md generic   Apr 21, 2015, 7:43:24 PM

123/1234/f.js typescript 1 Lines 59 KB Jan 15, 2019, 3:08:12 AM

123/1234/f.txt generic 59 KB Jan 15, 2019, 3:06:31 AM

123/1234/index.htm html 15 Lines 1.7 KB Jan 15, 2019, 4:20:05 AM

123/1234/jquery.wordexport.js typescript 49 Lines 4 KB Jan 26, 2021, 12:38:55 AM

C:/Users/Administrator/AppData/Local/Fortify/sca21.2/build/BD/extracted/javascript/F/BD/123/123/index.htm.js secondary   Mar 16, 2023, 4:38:42 PM

C:/Users/Administrator/AppData/Local/Fortify/sca21.2/build/BD/extracted/javascript/F/BD/123/1234/index.htm.js secondary   Mar 16, 2023, 4:38:42 PM

C:/Users/Administrator/AppData/Local/Fortify/sca21.2/build/BD/extracted/javascript/F/BD/auditor/   .html.js secondary 1.3 KB Mar 16, 2023, 4:38:42 PM

C:/Users/Administrator/AppData/Local/Fortify/sca21.2/build/BD/extracted/javascript/F/BD/chosen/index.html.js secondary   Mar 16, 2023, 4:38:42 PM

C:/Users/Administrator/AppData/Local/Fortify/sca21.2/build/BD/extracted/javascript/F/BD/chosen/index.proto.html.js secondary   Mar 16, 2023, 4:38:42 PM

C:/Users/Administrator/AppData/Local/Fortify/sca21.2/build/BD/extracted/javascript/F/BD/chosen/options.html.js secondary   Mar 16, 2023, 4:38:42 PM

C:/Users/Administrator/AppData/Local/Fortify/sca21.2/build/BD/extracted/javascript/F/BD/filter-table/filterTable.v1.0.html.js secondary 3.8 KB Mar 16, 2023, 4:38:42 PM

C:/Users/Administrator/AppData/Local/Fortify/sca21.2/build/BD/extracted/javascript/F/BD/mail/index.html.js secondary   Mar 16, 2023, 4:38:42 PM

C:/Users/Administrator/AppData/Local/Fortify/sca21.2/build/BD/extracted/javascript/F/BD/mail/thank-you.html.js secondary   Mar 16, 2023, 4:38:42 PM

C:/Users/Administrator/AppData/Local/Fortify/sca21.2/build/BD/extracted/javascript/F/BD/options.html.js secondary   Mar 16, 2023, 4:38:42 PM

C:/Users/Administrator/AppData/Local/Fortify/sca21.2/build/BD/extracted/javascript/F/BD/   .html.js secondary 1.3 KB Mar 16, 2023, 4:38:42 PM

File.php php 10 Lines 2.2 KB Mar 6, 2023, 1:53:09 AM

admin111.php php 19 Lines 2.4 KB Mar 3, 2023, 1:31:21 AM

adminpanel - Copy.php php 5 Lines 5.1 KB Oct 29, 2018, 5:20:40 AM

adminpanel.php php 5 Lines 7.2 KB Mar 6, 2023, 2:03:08 AM

adminpanel2.php php 5 Lines 5.1 KB Oct 29, 2018, 5:20:40 AM

adresspassilki - Copy.js typescript 22 Lines 1.9 KB Mar 23, 2021, 12:25:16 AM

Copyright 2021 Micro Focus or one of its affiliates.

adresspassilki.js typescript 22 Lines 1.9 KB Mar 23, 2021, 9:23:31 PM
adresspassilki.php php 1 Lines Mar 19, 2021, 2:52:08 AM
auditor/File.php php 10 Lines Oct 29, 2018, 9:22:07 PM
auditor/adminpanel.php php 5 Lines 3.8 KB Oct 29, 2018, 10:16:18 PM
auditor/adminpanel2.php php 5 Lines 5.1 KB Oct 29, 2018, 5:20:40 AM
auditor/filter-all1.php php 71 Lines 6.6 KB Oct 29, 2018, 10:21:44 PM
auditor/filter-all1seach.php php 71 Lines 6.5 KB Oct 29, 2018, 9:22:22 PM
auditor/filter-all2.php php 6 Lines 5.7 KB Oct 29, 2018, 10:13:53 PM
auditor/filter-all3.php php 22 Lines 6.7 KB Oct 29, 2018, 9:22:43 PM
auditor/filter-allold3.php php 22 Lines 6.7 KB Oct 29, 2018, 9:22:52 PM
auditor/img/redit.php php 887 Lines 86.1 KB Jan 29, 2017, 5:02:36 AM
auditor/img/redit1.php php 28 Lines 3 KB Jan 29, 2017, 5:02:36 AM
auditor/in.php php 19 Lines 2.3 KB Jun 18, 2019, 10:27:35 PM
auditor/jquery-1.12.2.min2.js typescript 5 Lines 95 KB Jan 29, 2017, 5:02:36 AM
auditor/js.js typescript 6 Lines Aug 7, 2018, 4:41:35 AM
auditor/msg-result.php php 1 Lines Jul 20, 2017, 9:18:44 PM
auditor/msg.php php 5 Lines Oct 29, 2018, 9:23:01 PM
auditor/msg01.php php 5 Lines Oct 29, 2018, 9:23:07 PM
auditor/msg15.php php 5 Lines Oct 29, 2018, 9:23:13 PM
auditor/script.js typescript 55 Lines 2.5 KB Sep 19, 2018, 10:50:04 PM
auditor/upload.php php 8 Lines Oct 29, 2018, 9:23:19 PM
auditor/upload1.php php 5 Lines Oct 29, 2018, 9:21:59 PM
auditor/ .html html 5 KB Oct 29, 2018, 7:54:01 PM
auditor/ _files/analytics.js. generic 42.2 KB Oct 29, 2018, 7:47:07 PM
auditor/.php php 20.8 KB Aug 14, 2018, 3:37:20 AM
chosen.css generic 11.9 KB Jun 8, 2018, 1:43:58 AM
chosen.jquery.js typescript 737 Lines 46.4 KB Jun 8, 2018, 1:43:58 AM
chosen.min.css generic 10 KB Jun 8, 2018, 1:43:58 AM
chosen.proto.js typescript 761 Lines 47.3 KB Jun 8, 2018, 1:43:58 AM
chosen/chosen.css generic 11.9 KB Jun 8, 2018, 1:43:58 AM
chosen/chosen.jquery.js typescript 737 Lines 46.4 KB Jun 8, 2018, 1:43:58 AM
chosen/chosen.min.css generic 10 KB Jun 8, 2018, 1:43:58 AM
chosen/chosen.proto.js typescript 761 Lines 47.3 KB Jun 8, 2018, 1:43:58 AM
chosen/composer.json json Jun 8, 2018, 1:43:58 AM
chosen/docsupport/chosen.jquery.js typescript 737 Lines 46.4 KB Jun 8, 2018, 1:43:58 AM
chosen/docsupport/init.js typescript 10 Lines Jun 8, 2018, 1:43:58 AM
chosen/docsupport/init.proto.js typescript 12 Lines Jun 8, 2018, 1:43:58 AM
chosen/docsupport/prism.js typescript 6 Lines 6.5 KB Jun 8, 2018, 1:43:58 AM
chosen/docsupport/prototype-1.7.0.0.js typescript 3,404 Lines 159.5 KB Jun 8, 2018, 1:43:58 AM
chosen/docsupport/send2.php php 1 Lines 1.3 KB Mar 22, 2021, 9:47:54 PM
chosen/index.html html 4 Lines 16.2 KB Mar 23, 2021, 10:40:49 PM
chosen/index.proto.html html 4 Lines 82.6 KB Mar 23, 2021, 10:12:49 PM
chosen/options.html html 1 Lines 13.8 KB Jun 8, 2018, 1:43:58 AM
chosen/send222.php php 1 Lines 16.2 KB Mar 23, 2021, 10:43:44 PM
composer.json json Jun 8, 2018, 1:43:58 AM
docsupport/chosen.jquery.js typescript 737 Lines 46.4 KB Jun 8, 2018, 1:43:58 AM
docsupport/init.js typescript 10 Lines Jun 8, 2018, 1:43:58 AM
docsupport/init.proto.js typescript 12 Lines Jun 8, 2018, 1:43:58 AM
docsupport/prism.js typescript 6 Lines 6.5 KB Jun 8, 2018, 1:43:58 AM
docsupport/prototype-1.7.0.0.js typescript 3,404 Lines 159.5 KB Jun 8, 2018, 1:43:58 AM
docsupport/send2.php php 1 Lines 1.3 KB Mar 22, 2021, 9:47:54 PM

Copyright 2021 Micro Focus or one of its affiliates.

download.php php 14 Lines 1.1 KB Jan 30, 2019, 9:11:49 PM
excel/Classes/PHPExcel.php php 190 Lines 18.9 KB May 19, 2012, 12:53:16 AM
excel/Classes/PHPExcel/Autoloader.php php 16 Lines 2.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/CachedObjectStorage/APC.php php 73 Lines 8.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/CachedObjectStorage/CacheBase.php php 38 Lines 6.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/CachedObjectStorage/DiscISAM.php php 50 Lines 6.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/CachedObjectStorage/ICache.php php 1 Lines 3.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/CachedObjectStorage/Igbinary.php php 26 Lines 4.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/CachedObjectStorage/Memcache.php php 78 Lines 9.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/CachedObjectStorage/Memory.php php 13 Lines 3.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/CachedObjectStorage/MemoryGZip.php php 24 Lines 3.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/CachedObjectStorage/MemorySerialized.php php 25 Lines 3.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/CachedObjectStorage/PHPTemp.php php 43 Lines 5.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/CachedObjectStorage/SQLite.php php 74 Lines 8.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/CachedObjectStorage/SQLite3.php php 80 Lines 8.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/CachedObjectStorage/Wincache.php php 79 Lines 8.6 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/CachedObjectStorageFactory.php php 69 Lines 6.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation.php php 1,316 Lines 167.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation/Database.php php 122 Lines 27.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation/DateTime.php php 600 Lines 53.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation/Engineering.php php 715 Lines 95.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation/Exception.php php 5 Lines 1.6 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation/ExceptionHandler.php php 3 Lines 1.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation/Financial.php php 813 Lines 83.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation/FormulaParser.php php 244 Lines 21.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation/FormulaToken.php php 35 Lines 5.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation/Function.php php 27 Lines 3.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation/Functions.php php 246 Lines 20 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation/Logical.php php 63 Lines 9.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation/LookupRef.php php 302 Lines 28.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation/MathTrig.php php 475 Lines 34.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation/Statistical.php php 1,340 Lines 106.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Calculation/TextData.php php 212 Lines 17.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Cell.php php 234 Lines 22.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Cell/AdvancedValueBinder.php php 47 Lines 5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Cell/DataType.php php 24 Lines 3.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Cell/DataValidation.php php 108 Lines 8.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Cell/DefaultValueBinder.php php 27 Lines 3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Cell/Hyperlink.php php 17 Lines 2.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Cell/IValueBinder.php php 1 Lines 1.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Chart.php php 107 Lines 9.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Chart/DataSeries.php php 87 Lines 7.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Chart/DataSeriesValues.php php 57 Lines 5.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Chart/Layout.php php 23 Lines 4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Chart/Legend.php php 36 Lines 4.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Chart/PlotArea.php php 17 Lines 2.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Chart/Renderer/jpgraph.php php 431 Lines 27.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Chart/Title.php php 10 Lines 1.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Comment.php php 58 Lines 6.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/DocumentProperties.php php 200 Lines 12 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/DocumentSecurity.php php 37 Lines 4.8 KB May 19, 2012, 12:53:14 AM

Copyright 2021 Micro Focus or one of its affiliates.

excel/Classes/PHPExcel/HashTable.php php 38 Lines 4.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/IComparable.php php 1 Lines 1.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/IOFactory.php php 72 Lines 7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/NamedRange.php php 49 Lines 5.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Reader/CSV.php php 135 Lines 11.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Reader/DefaultReadFilter.php php 6 Lines 1.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Reader/Excel2003XML.php php 383 Lines 28.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Reader/Excel2007.php php 814 Lines 83.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Reader/Excel2007/Chart.php php 269 Lines 16.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Reader/Excel2007/Theme.php php 13 Lines 2.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Reader/Excel5.php php 2,812 Lines 223.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Reader/Excel5/Escher.php php 204 Lines 18.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Reader/Gnumeric.php php 500 Lines 31.6 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Reader/IReadFilter.php php 1 Lines 1.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Reader/IReader.php php 1 Lines 1.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Reader/OOCalc.php php 275 Lines 22 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Reader/SYLK.php php 200 Lines 14.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/ReferenceHelper.php php 263 Lines 24.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/RichText.php php 40 Lines 4.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/RichText/ITextElement.php php 1 Lines 1.6 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/RichText/Run.php php 16 Lines 2.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/RichText/TextElement.php php 16 Lines 2.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Settings.php php 55 Lines 9.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/CodePage.php php 51 Lines 4.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/Date.php php 118 Lines 10.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/Drawing.php php 75 Lines 8.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/Escher.php php 5 Lines 2.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/Escher/DgContainer.php php 8 Lines 1.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/Escher/DgContainer/SpgrContainer.php php 11 Lines 2.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/Escher/DgContainer/SpgrContainer/SpContainer.php php 37 Lines 7.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/Escher/DggContainer.php php 14 Lines 4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/Escher/DggContainer/BstoreContainer.php php 4 Lines 1.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/Escher/DggContainer/BstoreContainer/BSE.php php 17 Lines 2.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/Escher/DggContainer/BstoreContainer/BSE/Blip.php php 5 Lines 2.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/Excel5.php php 83 Lines 11.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/File.php php 35 Lines 3.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/Font.php php 372 Lines 22.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/CholeskyDecomposition.php php 35 Lines 3.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/EigenvalueDecomposition.php php 276 Lines 22.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/LUDecomposition.php php 60 Lines 6 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/Matrix.php php 448 Lines 28.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/QRDecomposition.php php 55 Lines 5.6 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/SingularValueDecomposition.php php 156 Lines 12.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/docs/docs.php php 4 Lines May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/docs/download.php php 20 Lines 1.6 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/docs/example.php php 4 Lines 5.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/docs/includes/credits.php php 1 Lines May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/docs/includes/footer.php php 1 Lines May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/docs/includes/header.php php 1 Lines May 19, 2012, 12:53:14 AM

Copyright 2021 Micro Focus or one of its affiliates.

excel/Classes/PHPExcel/Shared/JAMA/docs/includes/navbar.php php 1 Lines May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/docs/index.php php 4 Lines 1.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/docs/package.php php 11 Lines May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/docs/test.php php 4 Lines 1.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/examples/LMQuadTest.php php 46 Lines 2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/examples/LagrangeInterpolation.php php 18 Lines 1.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/examples/LagrangeInterpolation2.php php 18 Lines 1.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/examples/LevenbergMarquardt.php php 35 Lines 5.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/examples/MagicSquareExample.php php 65 Lines 4.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/examples/Stats.php php 439 Lines 54.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/examples/benchmark.php php 146 Lines 6.6 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/examples/polyfit.php php 43 Lines 1.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/examples/tile.php php 27 Lines 1.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/tests/TestMatrix.php php 214 Lines 13.6 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/utils/Error.php php 17 Lines 3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/JAMA/utils/Maths.php php 9 Lines May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/OLE.php php 160 Lines 14.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/OLE/ChainedBlockStream.php php 46 Lines 6.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/OLE/PPS.php php 47 Lines 6.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/OLE/PPS/File.php php 8 Lines 2.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/OLE/PPS/Root.php php 177 Lines 13.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/OLERead.php php 103 Lines 9.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PCLZip/pclzip.lib.php php 1,507 Lines 198.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/2dbarcodes.php php 51 Lines 8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/barcodes.php php 955 Lines 58.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/config/lang/bra.php php 2 Lines 1.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/config/lang/eng.php php 2 Lines 1.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/config/lang/ger.php php 2 Lines 1.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/config/lang/ita.php php 2 Lines 1.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/config/tcpdf_config.php php 44 Lines 5.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/config/tcpdf_config_alt.php php 42 Lines 5.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/almohanad.php php 100 Lines 13.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/arialunicid0-chinese-simplified.php php 1,747 Lines 321.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/arialunicid0-chinese-traditional.php php 1,747 Lines 321.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/arialunicid0-japanese.php php 1,747 Lines 321.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/arialunicid0-korean.php php 1,747 Lines 321.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/arialunicid0.php php 1,747 Lines 321.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/chinese.php php 12 Lines 2.6 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/courier.php php 30 Lines 2.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavusans.php php 515 Lines 49.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavusansb.php php 502 Lines 48.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavusansbi.php php 456 Lines 44 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavusanscondensed.php php 515 Lines 49.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavusanscondensedb.php php 502 Lines 48.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavusanscondensedbi.php php 456 Lines 43.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavusanscondensedi.php php 469 Lines 45 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavusansi.php php 469 Lines 45.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavusansmono.php php 318 Lines 30 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavusansmonob.php php 305 Lines 28.8 KB May 19, 2012, 12:53:14 AM

Copyright 2021 Micro Focus or one of its affiliates.

excel/Classes/PHPExcel/Shared/PDF/fonts/dejavusansmonobi.php php 248 Lines 22.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavusansmonoi.php php 260 Lines 24 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavuserif.php php 311 Lines 29.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavuserifb.php php 297 Lines 28.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavuserifbi.php php 297 Lines 28.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavuserifcondensed.php php 311 Lines 29.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavuserifcondensedb.php php 297 Lines 28 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavuserifcondensedbi.php php 297 Lines 28.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavuserifcondensedi.php php 310 Lines 29.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/dejavuserifi.php php 310 Lines 29.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/freemono.php php 346 Lines 32.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/freemonob.php php 202 Lines 18.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/freemonobi.php php 151 Lines 13.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/freemonoi.php php 219 Lines 19.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/freesans.php php 309 Lines 28.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/freesansb.php php 234 Lines 21.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/freesansbi.php php 223 Lines 20.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/freesansi.php php 237 Lines 21.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/freeserif.php php 502 Lines 47.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/freeserifb.php php 280 Lines 25.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/freeserifbi.php php 236 Lines 21.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/freeserifi.php php 257 Lines 23.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/helvetica.php php 30 Lines 2.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/helveticab.php php 30 Lines 2.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/helveticabi.php php 30 Lines 2.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/helveticai.php php 30 Lines 2.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/hysmyeongjostdmedium.php php 24 Lines 1.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/kozgopromedium.php php 41 Lines 3.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/kozminproregular.php php 41 Lines 3.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/msungstdlight.php php 21 Lines 1.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/stsongstdlight.php php 20 Lines 1.6 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/symbol.php php 29 Lines 2.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/times.php php 30 Lines 2.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/timesb.php php 30 Lines 2.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/timesbi.php php 30 Lines 2.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/timesi.php php 30 Lines 2.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/uni2cid_ac15.php php 23,607 Lines 312.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/uni2cid_ag15.php php 30,209 Lines 401.6 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/uni2cid_aj16.php php 15,699 Lines 204.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/uni2cid_ak12.php php 17,486 Lines 228.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/utils/makeallttffonts.php php 5 Lines 2.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/utils/makefont.php php 278 Lines 16 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/zapfdingbats.php php 29 Lines 2.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/fonts/zarbold.php php 41 Lines 5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/htmlcolors.php php 1 Lines 5.6 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/pdf417.php php 600 Lines 54.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/qrcode.php php 1,005 Lines 81.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/tcpdf.crt generic 2.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/tcpdf.p12 generic 1.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/tcpdf.php php 10,117 Lines 914.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/PDF/unicode_data.php php 18,200 Lines 223 KB May 19, 2012, 12:53:14 AM

Copyright 2021 Micro Focus or one of its affiliates.

excel/Classes/PHPExcel/Shared/PasswordHasher.php php 14 Lines 2.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/String.php php 147 Lines 21 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/XMLWriter.php php 27 Lines 3.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/ZipArchive.php php 12 Lines 2.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/ZipStreamWrapper.php php 39 Lines 4.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/trend/bestFitClass.php php 131 Lines 10.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/trend/exponentialBestFitClass.php php 27 Lines 4.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/trend/linearBestFitClass.php php 13 Lines 3.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/trend/logarithmicBestFitClass.php php 19 Lines 3.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/trend/polynomialBestFitClass.php php 60 Lines 6.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/trend/powerBestFitClass.php php 29 Lines 4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Shared/trend/trendClass.php php 71 Lines 5.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Style.php php 178 Lines 18.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Style/Alignment.php php 123 Lines 10.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Style/Border.php php 121 Lines 8.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Style/Borders.php php 98 Lines 11.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Style/Color.php php 186 Lines 13.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Style/Conditional.php php 63 Lines 6.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Style/Fill.php php 99 Lines 9.6 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Style/Font.php php 160 Lines 13.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Style/NumberFormat.php php 253 Lines 19.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Style/Protection.php php 53 Lines 6.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet.php php 665 Lines 71.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet/BaseDrawing.php php 105 Lines 9.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet/CellIterator.php php 20 Lines 3.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet/ColumnDimension.php php 42 Lines 5.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet/Drawing.php php 26 Lines 3.6 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet/Drawing/Shadow.php php 58 Lines 5.8 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet/HeaderFooter.php php 69 Lines 12.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet/HeaderFooterDrawing.php php 74 Lines 7.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet/MemoryDrawing.php php 43 Lines 4.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet/PageMargins.php php 30 Lines 4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet/PageSetup.php php 208 Lines 23.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet/Protection.php php 108 Lines 10 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet/Row.php php 8 Lines 2.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet/RowDimension.php php 37 Lines 4.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet/RowIterator.php php 19 Lines 3.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Worksheet/SheetView.php php 20 Lines 3.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/WorksheetIterator.php php 10 Lines 2.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/CSV.php php 66 Lines 6.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel2007.php php 154 Lines 18.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel2007/Chart.php php 620 Lines 34.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel2007/Comments.php php 105 Lines 8.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel2007/ContentTypes.php php 85 Lines 8.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel2007/DocProps.php php 110 Lines 8.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel2007/Drawing.php php 252 Lines 19.3 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel2007/Rels.php php 122 Lines 12.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel2007/StringTable.php php 125 Lines 10.1 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel2007/Style.php php 269 Lines 22.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel2007/Theme.php php 387 Lines 24.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel2007/Workbook.php php 153 Lines 14.5 KB May 19, 2012, 12:53:14 AM

Copyright 2021 Micro Focus or one of its affiliates.

excel/Classes/PHPExcel/Writer/Excel2007/Worksheet.php php 491 Lines 39.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel2007/WriterPart.php php 9 Lines 2.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel5.php php 380 Lines 31 KB May 19, 2012, 12:53:16 AM
excel/Classes/PHPExcel/Writer/Excel5/BIFFwriter.php php 51 Lines 7.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel5/Escher.php php 247 Lines 12.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel5/Font.php php 47 Lines 3.9 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel5/Parser.php php 665 Lines 55.4 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel5/Workbook.php php 527 Lines 44.2 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel5/Worksheet.php php 1,105 Lines 84.7 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/Excel5/Xf.php php 127 Lines 16.5 KB May 19, 2012, 12:53:14 AM
excel/Classes/PHPExcel/Writer/HTML.php php 441 Lines 38.7 KB May 19, 2012, 12:53:16 AM
excel/Classes/PHPExcel/Writer/IWriter.php php 1 Lines 1.4 KB May 19, 2012, 12:53:16 AM
excel/Classes/PHPExcel/Writer/PDF.php php 111 Lines 13.6 KB May 19, 2012, 12:53:16 AM
excel/config.php php 10 Lines Jul 29, 2012, 9:43:36 AM
excel/functions.php php 8 Lines Jul 29, 2012, 9:45:58 AM
excel/index.php php 97 Lines 5.4 KB Jul 30, 2012, 11:16:00 AM
excel/price.sql plsql 1 Lines 2 KB Jul 30, 2012, 12:54:30 PM
export/class/acontroller.php php 2 Lines Sep 29, 2015, 12:10:17 AM
export/class/aview.php php 8 Lines Aug 28, 2015, 5:21:48 AM
export/class/controller.php php 29 Lines 3.1 KB Aug 4, 2019, 9:21:54 PM
export/class/model.php php 24 Lines 1.9 KB Feb 13, 2020, 8:58:01 PM
export/class/view.php php 5 Lines Jan 28, 2019, 12:06:17 AM
export/composer.json json Sep 30, 2015, 5:27:33 AM
export/composer.lock generic 2.4 KB Sep 30, 2015, 5:27:38 AM
export/config.php php 13 Lines 1.4 KB Jan 24, 2021, 11:43:26 PM
export/index.php php 16 Lines Jan 24, 2021, 11:47:48 PM
export/indexclean.php php 12 Lines 2.9 KB Jan 25, 2021, 1:41:43 AM
export/main.sql plsql Jan 27, 2019, 8:37:59 PM
export/msg15.php php 5 Lines Jan 25, 2021, 12:27:54 AM
export/template/index.tpl.php php 1 Lines Jan 31, 2019, 11:43:29 PM
export/vendor/autoload.php php 2 Lines Sep 30, 2015, 5:27:38 AM
export/vendor/composer/ClassLoader.php php 114 Lines 12.1 KB Sep 30, 2015, 5:27:38 AM
export/vendor/composer/autoload_classmap.php php 3 Lines Sep 30, 2015, 5:27:38 AM
export/vendor/composer/autoload_namespaces.php php 4 Lines Sep 30, 2015, 5:27:38 AM
export/vendor/composer/autoload_psr4.php php 3 Lines Sep 30, 2015, 5:27:38 AM
export/vendor/composer/autoload_real.php php 20 Lines 1.4 KB Sep 30, 2015, 5:27:38 AM
export/vendor/composer/installed.json json 1.7 KB Sep 30, 2015, 5:27:38 AM
export/vendor/phpoffice/phpexcel/.travis.yml yaml Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel.php php 277 Lines 29.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Autoloader.php php 17 Lines 3.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/APC.php php 76 Lines 9.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/CacheBase.php php 74 Lines 10 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/DiscISAM.php php 51 Lines 6.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/ICache.php php 1 Lines 3.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/Igbinary.php php 29 Lines 4.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/Memcache.php php 81 Lines 9.7 KB Sep 30, 2015,

5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/Memory.php php 16 Lines 3.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/MemoryGZip.php php 27 Lines 4.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/MemorySerialized.php php 28 Lines 4.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/PHPTemp.php php 44 Lines 6.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/SQLite.php php 91 Lines 9.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/SQLite3.php php 100 Lines 10.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/Wincache.php php 82 Lines 9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorageFactory.php php 68 Lines 8.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CalcEngine/CyclicReferenceStack.php php 6 Lines 2.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/CalcEngine/Logger.php php 22 Lines 3.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation.php php 1,338 Lines 172.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Database.php php 122 Lines 27.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/DateTime.php php 616 Lines 54.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Engineering.php php 709 Lines 96.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Exception.php php 5 Lines 1.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/ExceptionHandler.php php 3 Lines 1.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Financial.php php 815 Lines 85.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/FormulaParser.php php 244 Lines 21.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/FormulaToken.php php 35 Lines 5.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Function.php php 27 Lines 3.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Functions.php php 191 Lines 17 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Logical.php php 63 Lines 10 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/LookupRef.php php 334 Lines 31.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/MathTrig.php php 495 Lines 37.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Statistical.php php 1,340 Lines 106.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/TextData.php php 234 Lines 18.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Token/Stack.php php 17 Lines 2.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Cell.php php 274 Lines 27.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Cell/AdvancedValueBinder.php php 81 Lines 8.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Cell/DataType.php php 24 Lines 3.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Cell/DataValidation.php php 108 Lines 10.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Cell/DefaultValueBinder.php php 36 Lines 3.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Cell/Hyperlink.php php 17 Lines 2.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Cell/IValueBinder.php php 1 Lines 1.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart.php php 135 Lines 12.5 KB Sep 30, 2015, 5:27:36 AM

Copyright 2021 Micro Focus or one of its affiliates.

export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/Axis.php php 94 Lines 14.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/DataSeries.php php 103 Lines 8.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/DataSeriesValues.php php 87 Lines 7.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/Exception.php php 5 Lines 1.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/GridLines.php php 95 Lines 11.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/Layout.php php 58 Lines 8.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/Legend.php php 39 Lines 4.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/PlotArea.php php 18 Lines 2.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/Properties.php php 59 Lines 9.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/Renderer/jpgraph.php php 447 Lines 29.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/Title.php php 11 Lines 2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Comment.php php 59 Lines 6.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/DocumentProperties.php php 200 Lines 15.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/DocumentSecurity.php php 37 Lines 4.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Exception.php php 5 Lines 1.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/HashTable.php php 38 Lines 4.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Helper/HTML.php php 91 Lines 23.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/IComparable.php php 1 Lines 1.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/IOFactory.php php 86 Lines 7.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/NamedRange.php php 49 Lines 5.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Abstract.php php 41 Lines 6.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/CSV.php php 105 Lines 8.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/DefaultReadFilter.php php 6 Lines 1.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel2003XML.php php 370 Lines 27.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel2007.php php 898 Lines 95.6 KB Jan 28, 2019, 12:06:34 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel2007/Chart.php php 318 Lines 19.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel2007/Theme.php php 13 Lines 2.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel5.php php 2,939 Lines 232 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel5/Escher.php php 204 Lines 18.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel5/MD5.php php 100 Lines 8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel5/RC4.php php 13 Lines 2.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Exception.php php 5 Lines 1.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Gnumeric.php php 495 Lines 29.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/HTML.php php 209 Lines 20 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/IReadFilter.php php 1 Lines 1.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/IReader.php php 1 Lines 1.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/OOCalc.php php 317 Lines 25.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/SYLK.php php 194 Lines 13.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/ReferenceHelper.php php 351 Lines 39 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/RichText.php php 40 Lines 5.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/RichText/ITextElement.php php 1 Lines 1.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/RichText/Run.php php 16 Lines 2.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/RichText/TextElement.php php 16 Lines 2.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Settings.php php 71 Lines 12.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/CodePage.php php 56 Lines 4.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Date.php php 141 Lines 12.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Drawing.php php 75 Lines 8.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher.php php 5 Lines 2.2 KB Sep 30, 2015, 5:27:36 AM

Copyright 2021 Micro Focus or one of its affiliates.

export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher/DgContainer.php php 8 Lines 1.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher/DgContainer/SpgrContainer.php php 11 Lines 2.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher/DgContainer/SpgrContainer/SpContainer.php php 39 Lines 7.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher/DggContainer.php php 14 Lines 4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher/DggContainer/BstoreContainer.php php 4 Lines 1.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher/DggContainer/BstoreContainer/BSE.php php 17 Lines 2.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher/DggContainer/BstoreContainer/BSE/Blip.php php 5 Lines 2.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Excel5.php php 83 Lines 11.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/File.php php 44 Lines 5.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Font.php php 375 Lines 22.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/CholeskyDecomposition.php php 35 Lines 3.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/EigenvalueDecomposition.php php 276 Lines 22.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/LUDecomposition.php php 60 Lines 6.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/Matrix.php php 448 Lines 29.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/QRDecomposition.php php 55 Lines 5.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/SingularValueDecomposition.php php 156 Lines 12.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/utils/Error.php php 17 Lines 3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/utils/Maths.php php 9 Lines Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/OLE.php php 160 Lines 14.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/OLE/ChainedBlockStream.php php 42 Lines 5.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/OLE/PPS.php php 47 Lines 6.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/OLE/PPS/File.php php 8 Lines 2.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/OLE/PPS/Root.php php 177 Lines 13.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/OLERead.php php 108 Lines 9.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/PCLZip/pclzip.lib.php php 1,507 Lines 198.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/PasswordHasher.php php 13 Lines 2.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/String.php php 170 Lines 23.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/TimeZone.php php 26 Lines 4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/XMLWriter.php php 29 Lines 3.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/ZipArchive.php php 41 Lines 5.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/ZipStreamWrapper.php php 41 Lines 5.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/trend/bestFitClass.php php 131 Lines 10.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/trend/exponentialBestFitClass.php php 27 Lines 4.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/trend/linearBestFitClass.php php 13 Lines 3.3 KB Sep 30, 2015,

Copyright 2021 Micro Focus or one of its affiliates.

5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/trend/logarithmicBestFitClass.php php 19 Lines 3.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/trend/polynomialBestFitClass.php php 60 Lines 6.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/trend/powerBestFitClass.php php 29 Lines 4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/trend/trendClass.php php 71 Lines 5.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style.php php 181 Lines 23.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Alignment.php php 132 Lines 10.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Border.php php 98 Lines 7.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Borders.php php 87 Lines 10.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Color.php php 170 Lines 12.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Conditional.php php 63 Lines 6.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Fill.php php 90 Lines 8.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Font.php php 159 Lines 12.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/NumberFormat.php php 281 Lines 20.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Protection.php php 41 Lines 5.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Supervisor.php php 15 Lines 3.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet.php php 723 Lines 90.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/AutoFilter.php php 348 Lines 31 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/AutoFilter/Column.php php 78 Lines 9.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/AutoFilter/Column/Rule.php php 189 Lines 16.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/BaseDrawing.php php 107 Lines 10.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/CellIterator.php php 8 Lines 2.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/Column.php php 8 Lines 2.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/ColumnCellIterator.php php 48 Lines 6.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/ColumnDimension.php php 42 Lines 5.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/ColumnIterator.php php 34 Lines 5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/Drawing.php php 26 Lines 3.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/Drawing/Shadow.php php 58 Lines 5.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/HeaderFooter.php php 69 Lines 12.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/HeaderFooterDrawing.php php 74 Lines 7.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/MemoryDrawing.php php 43 Lines 4.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/PageMargins.php php 30 Lines 4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/PageSetup.php php 208 Lines 23.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/Protection.php php 108 Lines 10 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/Row.php php 9 Lines 2.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/RowCellIterator.php php 53 Lines 6.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/RowDimension.php php 42 Lines 5.1 KB Sep 30, 2015, 5:27:36 AM

Copyright 2021 Micro Focus or one of its affiliates.

export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/RowIterator.php php 29 Lines 4.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/SheetView.php php 36 Lines 4.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/WorksheetIterator.php php 10 Lines 2.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Abstract.php php 22 Lines 4.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/CSV.php php 72 Lines 7.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007.php php 156 Lines 18.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Chart.php php 930 Lines 56.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Comments.php php 105 Lines 8.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/ContentTypes.php php 95 Lines 10.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/DocProps.php php 110 Lines 8.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Drawing.php php 252 Lines 19.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Rels.php php 129 Lines 13.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/RelsRibbon.php php 19 Lines 2.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/RelsVBA.php php 16 Lines 2.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/StringTable.php php 128 Lines 10.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Style.php php 294 Lines 24.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Theme.php php 387 Lines 24.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Workbook.php php 155 Lines 15.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Worksheet.php php 542 Lines 43.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/WriterPart.php php 9 Lines 2.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5.php php 413 Lines 33 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5/BIFFwriter.php php 51 Lines 7.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5/Escher.php php 255 Lines 13.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5/Font.php php 47 Lines 3.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5/Parser.php php 666 Lines 55.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5/Workbook.php php 550 Lines 45.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5/Worksheet.php php 1,674 Lines 121.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5/Xf.php php 127 Lines 16.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Exception.php php 5 Lines 1.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/HTML.php php 532 Lines 45.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/IWriter.php php 1 Lines 1.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument.php php 50 Lines 6.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/Cell/Comment.php php 13 Lines 2.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/Content.php php 147 Lines 12.6 KB Sep 30, 2015, 5:27:36 AM

Copyright 2021 Micro Focus or one of its affiliates.

export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/Meta.php php 39 Lines 4.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/MetaInf.php php 39 Lines 4.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/Mimetype.php php 3 Lines 1.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/Settings.php php 28 Lines 3.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/Styles.php php 44 Lines 5.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/Thumbnails.php php 3 Lines 1.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/WriterPart.php php 1 Lines 1.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/PDF.php php 19 Lines 3.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/PDF/Core.php php 80 Lines 13.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/PDF/DomPDF.php php 38 Lines 4.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/PDF/mPDF.php php 46 Lines 4.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/PDF/tcPDF.php php 47 Lines 5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/01pharSimple.php php 49 Lines 4.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/01simple-download-ods.php php 36 Lines 3.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/01simple-download-pdf.php php 37 Lines 3.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/01simple-download-xls.php php 36 Lines 3.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/01simple-download-xlsx.php php 36 Lines 3.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/01simple.php php 52 Lines 4.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/01simplePCLZip.php php 44 Lines 3.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/02types-xls.php php 100 Lines 7.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/02types.php php 99 Lines 7.6 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/03formulas.php php 72 Lines 6.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/04printing.php php 56 Lines 5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/05featuredemo.inc.php php 226 Lines 18.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/05featuredemo.php php 28 Lines 2.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/06largescale-with-cellcaching-sqlite.php php 57 Lines 4.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/06largescale-with-cellcaching-sqlite3.php php 57 Lines 4.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/06largescale-with-cellcaching.php php 57 Lines 4.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/06largescale-xls.php php 53 Lines 4.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/06largescale.php php 53 Lines 4.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/07reader.php php 27 Lines 2.7 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/07readerPCLZip.php php 28 Lines 2.9 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/08conditionalformatting.php php 98 Lines 8.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/08conditionalformatting2.php php 69 Lines 5.2 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/09pagebreaks.php php 62 Lines 5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/10autofilter-selection-1.php php 120 Lines 8.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/10autofilter-selection-2.php php 119 Lines 8.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/10autofilter-selection-display.php php 108 Lines 7.4 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/10autofilter.php php 105 Lines 7.5 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/11documentsecurity-xls.php php 47 Lines 4.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/11documentsecurity.php php 47 Lines 4.1 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/12cellProtection.php php 44 Lines 3.7 KB Sep 30, 2015, 5:27:36 AM

Copyright 2021 Micro Focus or one of its affiliates.

export/vendor/phpoffice/phpexcel/Examples/13calculation.php php 140 Lines 10.8 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/14excel5.php php 19 Lines 2.3 KB Sep 30, 2015, 5:27:36 AM
export/vendor/phpoffice/phpexcel/Examples/15datavalidation-xls.php php 81 Lines 6 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/15datavalidation.php php 81 Lines 6.1 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/16csv.php php 53 Lines 4.6 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/17html.php php 21 Lines 2.4 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/18extendedcalculation.php php 54 Lines 4.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/19namedrange.php php 61 Lines 5.2 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/20readexcel5.php php 28 Lines 2.9 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/21pdf.php php 31 Lines 3.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/22heavilyformatted.php php 49 Lines 4.2 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/23sharedstyles.php php 56 Lines 4.5 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/24readfilter.php php 26 Lines 2.5 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/25inmemoryimage.php php 36 Lines 3.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/26utf8.php php 47 Lines 4.8 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/27imagesexcel5.php php 20 Lines 2.4 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/28iterator.php php 22 Lines 2.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/29advancedvaluebinder.php php 96 Lines 7.5 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/30template.php php 31 Lines 3.1 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/31docproperties_write-xls.php php 57 Lines 4.9 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/31docproperties_write.php php 57 Lines 5 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/32chartreadwrite.php php 63 Lines 4.4 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/33chartcreate-area.php php 47 Lines 4.5 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/33chartcreate-bar-stacked.php php 48 Lines 4.7 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/33chartcreate-bar.php php 48 Lines 4.7 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/33chartcreate-column-2.php php 57 Lines 5.2 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/33chartcreate-column.php php 48 Lines 4.7 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/33chartcreate-composite.php php 67 Lines 6.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/33chartcreate-line.php php 47 Lines 4.5 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/33chartcreate-multiple-charts.php php 71 Lines 7 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/33chartcreate-pie.php php 65 Lines 6.4 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/33chartcreate-radar.php php 55 Lines 4.9 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/33chartcreate-scatter.php php 48 Lines 4.5 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/33chartcreate-stock.php php 55 Lines 5.1 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/34chartupdate.php php 29 Lines 2.6 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/35chartrender.php php 54 Lines 4.1 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/36chartreadwriteHTML.php php 69 Lines 4.9 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/36chartreadwritePDF.php php 75 Lines 5.6 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/37page_layout_view.php php 33 Lines 3.2 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/38cloneWorksheet.php php 53 Lines 4.1 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/39dropdown.php php 99 Lines 5.8 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/40duplicateStyle.php php 32 Lines 1.7 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/41password.php php 31 Lines 3.1 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/42richText.php php 65 Lines 5.2 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/Excel2003XMLReader.php php 17 Lines 2.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/Excel2003XMLTest.xml xml 937 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/GnumericReader.php php 17 Lines 2.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/OOCalcReader.php php 17 Lines 2.2 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/OOCalcReaderPCLZip.php php 18 Lines 2.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/Quadratic.php php 16 Lines 2.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/Quadratic2.php php 13 Lines 2.4 KB Sep 30, 2015, 5:27:37 AM

Copyright 2021 Micro Focus or one of its affiliates.

export/vendor/phpoffice/phpexcel/Examples/SylkReader.php php 11 Lines 1.9 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/XMLReader.php php 15 Lines 2.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/XMLTest.xml xml 20.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/Examples/runall.php php 91 Lines 3.7 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/composer.json json Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/AutoloaderTest.php php 18 Lines 1.5 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/DateTimeTest.php php 169 Lines 15 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/EngineeringTest.php php 236 Lines 19.6 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/FinancialTest.php php 161 Lines 13.7 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/FunctionsTest.php php 87 Lines 7.1 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/LogicalTest.php php 35 Lines 2.7 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/LookupRefTest.php php 16 Lines 1.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/MathTrigTest.php php 201 Lines 14.9 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/TextDataTest.php php 117 Lines 9.7 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/CalculationTest.php php 13 Lines 1.4 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Cell/AdvancedValueBinderTest.php php 45 Lines 2.8 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Cell/DataTypeTest.php php 8 Lines Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Cell/DefaultValueBinderTest.php php 38 Lines 2.2 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Cell/HyperlinkTest.php php 40 Lines 2.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/CellTest.php php 94 Lines 8.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Chart/DataSeriesValuesTest.php php 20 Lines 1.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Chart/LayoutTest.php php 13 Lines Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Chart/LegendTest.php php 62 Lines 3.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Reader/XEEValidatorTest.php php 19 Lines 1.4 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/ReferenceHelperTest.php php 35 Lines 1.3 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Shared/CodePageTest.php php 18 Lines 1.5 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Shared/DateTest.php php 67 Lines 5.2 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Shared/FileTest.php php 15 Lines Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Shared/FontTest.php php 32 Lines 2.5 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Shared/PasswordHasherTest.php php 10 Lines Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Shared/StringTest.php php 30 Lines 2.4 KB Sep 30, 2015, 5:27:37 AM

Copyright 2021 Micro Focus or one of its affiliates.

export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Shared/TimeZoneTest.php php 16 Lines Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Style/ColorTest.php php 25 Lines 2 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Style/NumberFormatTest.php php 12 Lines Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/AutoFilter/Column/RuleTest.php php 41 Lines 3.6 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/AutoFilter/ColumnTest.php php 60 Lines 5.2 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/AutoFilterTest.php php 128 Lines 10.4 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/CellCollectionTest.php php 14 Lines Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/ColumnCellIteratorTest.php php 39 Lines 2.7 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/ColumnIteratorTest.php php 40 Lines 2.8 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/RowCellIteratorTest.php php 40 Lines 2.8 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/RowIteratorTest.php php 39 Lines 2.5 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/WorksheetColumnTest.php php 21 Lines 1.4 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/WorksheetRowTest.php php 21 Lines 1.4 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/bootstrap.php php 18 Lines 1.4 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/custom/Complex.php php 54 Lines 3.6 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/custom/complexAssert.php php 34 Lines 2 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/phpunit-cc.xml xml 1.5 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/phpunit.xml xml 1 KB Sep 30, 2015, 5:27:37 AM
export/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestInvalidUTF-16.xml xml Sep 30, 2015, 5:27:38 AM
export/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestInvalidUTF-16BE.xml xml Sep 30, 2015, 5:27:38 AM
export/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestInvalidUTF-16LE.xml xml Sep 30, 2015, 5:27:38 AM
export/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestInvalidUTF-8.xml xml Sep 30, 2015, 5:27:38 AM
export/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestValidUTF-16.xml xml Sep 30, 2015, 5:27:38 AM
export/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestValidUTF-16BE.xml xml Sep 30, 2015, 5:27:38 AM
export/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestValidUTF-16LE.xml xml Sep 30, 2015, 5:27:38 AM
export/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestValidUTF-8.xml xml Sep 30, 2015, 5:27:38 AM
export/vendor/phpoffice/phpexcel/unitTests/testDataFileIterator.php php 57 Lines 3.8 KB Sep 30, 2015, 5:27:38 AM
exportold/class/acontroller.php php 2 Lines Sep 29, 2015, 12:10:17 AM
exportold/class/aview.php php 8 Lines Aug 28, 2015, 5:21:48 AM
exportold/class/controller.php php 29 Lines 3.1 KB Aug 4, 2019, 9:21:54 PM
exportold/class/model.php php 24 Lines 1.9 KB Jan 24, 2020, 2:52:37 AM
exportold/class/view.php php 5 Lines Jan 28, 2019, 12:06:17 AM
exportold/composer.json json Sep 30, 2015, 5:27:33 AM
exportold/composer.lock generic 2.4 KB Sep 30, 2015, 5:27:38 AM
exportold/config.php php 13 Lines 1.4 KB Jan 24, 2021, 11:42:49 PM
exportold/index.php php 16 Lines Jan 24, 2021, 11:48:48 PM
exportold/indexcleanold.php php 12 Lines 2.9 KB Jan 25, 2021, 1:32:16 AM
exportold/main.sql plsql Jan 27, 2019, 8:37:59 PM
exportold/msg15.php php 5 Lines Jan 25, 2021, 12:27:20 AM
exportold/template/index.tpl.php php 1 Lines Jan 31, 2019, 11:43:29 PM

Copyright 2021 Micro Focus or one of its affiliates.

exportold/vendor/autoload.php php 2 Lines Sep 30, 2015, 5:27:38 AM
exportold/vendor/composer/ClassLoader.php php 114 Lines 12.1 KB Sep 30, 2015, 5:27:38 AM
exportold/vendor/composer/autoload_classmap.php php 3 Lines Sep 30, 2015, 5:27:38 AM
exportold/vendor/composer/autoload_namespaces.php php 4 Lines Sep 30, 2015, 5:27:38 AM
exportold/vendor/composer/autoload_psr4.php php 3 Lines Sep 30, 2015, 5:27:38 AM
exportold/vendor/composer/autoload_real.php php 20 Lines 1.4 KB Sep 30, 2015, 5:27:38 AM
exportold/vendor/composer/installed.json json 1.7 KB Sep 30, 2015, 5:27:38 AM
exportold/vendor/phpoffice/phpexcel/.travis.yml yaml Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel.php php 277 Lines 29.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Autoloader.php php 17 Lines 3.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/APC.php php 76 Lines 9.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/CacheBase.php php 74 Lines 10 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/DiscISAM.php php 51 Lines 6.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/ICache.php php 1 Lines 3.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/Igbinary.php php 29 Lines 4.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/Memcache.php php 81 Lines 9.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/Memory.php php 16 Lines 3.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/MemoryGZip.php php 27 Lines 4.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/MemorySerialized.php php 28 Lines 4.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/PHPTemp.php php 44 Lines 6.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/SQLite.php php 91 Lines 9.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/SQLite3.php php 100 Lines 10.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorage/Wincache.php php 82 Lines 9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CachedObjectStorageFactory.php php 68 Lines 8.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CalcEngine/CyclicReferenceStack.php php 6 Lines 2.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/CalcEngine/Logger.php php 22 Lines 3.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation.php php 1,338 Lines 172.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Database.php php 122 Lines 27.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/DateTime.php php 616 Lines 54.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Engineering.php php 709 Lines 96.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Exception.php php 5 Lines 1.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/ExceptionHandler.php php 3 Lines 1.5 KB Sep 30, 2015, 5:27:36 AM

Copyright 2021 Micro Focus or one of its affiliates.

exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Financial.php php 815 Lines 85.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/FormulaParser.php php 244 Lines 21.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/FormulaToken.php php 35 Lines 5.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Function.php php 27 Lines 3.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Functions.php php 191 Lines 17 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Logical.php php 63 Lines 10 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/LookupRef.php php 334 Lines 31.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/MathTrig.php php 495 Lines 37.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Statistical.php php 1,340 Lines 106.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/TextData.php php 234 Lines 18.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Calculation/Token/Stack.php php 17 Lines 2.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Cell.php php 274 Lines 27.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Cell/AdvancedValueBinder.php php 81 Lines 8.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Cell/DataType.php php 24 Lines 3.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Cell/DataValidation.php php 108 Lines 10.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Cell/DefaultValueBinder.php php 36 Lines 3.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Cell/Hyperlink.php php 17 Lines 2.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Cell/IValueBinder.php php 1 Lines 1.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart.php php 135 Lines 12.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/Axis.php php 94 Lines 14.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/DataSeries.php php 103 Lines 8.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/DataSeriesValues.php php 87 Lines 7.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/Exception.php php 5 Lines 1.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/GridLines.php php 95 Lines 11.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/Layout.php php 58 Lines 8.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/Legend.php php 39 Lines 4.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/PlotArea.php php 18 Lines 2.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/Properties.php php 59 Lines 9.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/Renderer/jpgraph.php php 447 Lines 29.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Chart/Title.php php 11 Lines 2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Comment.php php 59 Lines 6.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/DocumentProperties.php php 200 Lines 15.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/DocumentSecurity.php php 37 Lines 4.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Exception.php php 5 Lines 1.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/HashTable.php php 38 Lines 4.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Helper/HTML.php php 91 Lines 23.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/IComparable.php php 1 Lines 1.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/IOFactory.php php 86 Lines 7.8 KB Sep 30, 2015, 5:27:36 AM

Copyright 2021 Micro Focus or one of its affiliates.

exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/NamedRange.php php 49 Lines 5.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Abstract.php php 41 Lines 6.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/CSV.php php 105 Lines 8.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/DefaultReadFilter.php php 6 Lines 1.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel2003XML.php php 370 Lines 27.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel2007.php php 898 Lines 95.6 KB Jan 28, 2019, 12:06:34 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel2007/Chart.php php 318 Lines 19.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel2007/Theme.php php 13 Lines 2.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel5.php php 2,939 Lines 232 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel5/Escher.php php 204 Lines 18.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel5/MD5.php php 100 Lines 8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Excel5/RC4.php php 13 Lines 2.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Exception.php php 5 Lines 1.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/Gnumeric.php php 495 Lines 29.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/HTML.php php 209 Lines 20 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/IReadFilter.php php 1 Lines 1.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/IReader.php php 1 Lines 1.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/OOCalc.php php 317 Lines 25.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Reader/SYLK.php php 194 Lines 13.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/ReferenceHelper.php php 351 Lines 39 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/RichText.php php 40 Lines 5.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/RichText/ITextElement.php php 1 Lines 1.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/RichText/Run.php php 16 Lines 2.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/RichText/TextElement.php php 16 Lines 2.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Settings.php php 71 Lines 12.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/CodePage.php php 56 Lines 4.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Date.php php 141 Lines 12.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Drawing.php php 75 Lines 8.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher.php php 5 Lines 2.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher/DgContainer.php php 8 Lines 1.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher/DgContainer/SpgrContainer.php php 11 Lines 2.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher/DgContainer/SpgrContainer/SpContainer.php php 39 Lines 7.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher/DggContainer.php php 14 Lines 4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher/DggContainer/BstoreContainer.php php 4 Lines 1.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher/DggContainer/BstoreContainer/BSE.php php 17 Lines 2.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Escher/DggContainer/BstoreContainer/BSE/Blip.php php 5 Lines 2.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Excel5.php php 83 Lines 11.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/File.php php 44 Lines 5.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/Font.php php 375 Lines 22.8 KB Sep 30, 2015, 5:27:36 AM

Copyright 2021 Micro Focus or one of its affiliates.

exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/CholeskyDecomposition.php php 35 Lines 3.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/EigenvalueDecomposition.php php 276 Lines 22.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/LUDecomposition.php php 60 Lines 6.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/Matrix.php php 448 Lines 29.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/QRDecomposition.php php 55 Lines 5.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/SingularValueDecomposition.php php 156 Lines 12.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/utils/Error.php php 17 Lines 3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/JAMA/utils/Maths.php php 9 Lines Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/OLE.php php 160 Lines 14.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/OLE/ChainedBlockStream.php php 42 Lines 5.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/OLE/PPS.php php 47 Lines 6.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/OLE/PPS/File.php php 8 Lines 2.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/OLE/PPS/Root.php php 177 Lines 13.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/OLERead.php php 108 Lines 9.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/PCLZip/pclzip.lib.php php 1,507 Lines 198.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/PasswordHasher.php php 13 Lines 2.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/String.php php 170 Lines 23.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/TimeZone.php php 26 Lines 4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/XMLWriter.php php 29 Lines 3.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/ZipArchive.php php 41 Lines 5.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/ZipStreamWrapper.php php 41 Lines 5.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/trend/bestFitClass.php php 131 Lines 10.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/trend/exponentialBestFitClass.php php 27 Lines 4.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/trend/linearBestFitClass.php php 13 Lines 3.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/trend/logarithmicBestFitClass.php php 19 Lines 3.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/trend/polynomialBestFitClass.php php 60 Lines 6.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/trend/powerBestFitClass.php php 29 Lines 4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Shared/trend/trendClass.php php 71 Lines 5.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style.php php 181 Lines 23.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Alignment.php php 132 Lines 10.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Border.php php 98 Lines 7.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Borders.php php 87 Lines 10.9 KB Sep 30, 2015, 5:27:36 AM

Copyright 2021 Micro Focus or one of its affiliates.

exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Color.php php 170 Lines 12.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Conditional.php php 63 Lines 6.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Fill.php php 90 Lines 8.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Font.php php 159 Lines 12.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/NumberFormat.php php 281 Lines 20.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Protection.php php 41 Lines 5.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Style/Supervisor.php php 15 Lines 3.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet.php php 723 Lines 90.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/AutoFilter.php php 348 Lines 31 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/AutoFilter/Column.php php 78 Lines 9.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/AutoFilter/Column/Rule.php php 189 Lines 16.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/BaseDrawing.php php 107 Lines 10.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/CellIterator.php php 8 Lines 2.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/Column.php php 8 Lines 2.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/ColumnCellIterator.php php 48 Lines 6.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/ColumnDimension.php php 42 Lines 5.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/ColumnIterator.php php 34 Lines 5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/Drawing.php php 26 Lines 3.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/Drawing/Shadow.php php 58 Lines 5.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/HeaderFooter.php php 69 Lines 12.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/HeaderFooterDrawing.php php 74 Lines 7.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/MemoryDrawing.php php 43 Lines 4.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/PageMargins.php php 30 Lines 4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/PageSetup.php php 208 Lines 23.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/Protection.php php 108 Lines 10 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/Row.php php 9 Lines 2.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/RowCellIterator.php php 53 Lines 6.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/RowDimension.php php 42 Lines 5.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/RowIterator.php php 29 Lines 4.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Worksheet/SheetView.php php 36 Lines 4.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/WorksheetIterator.php php 10 Lines 2.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Abstract.php php 22 Lines 4.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/CSV.php php 72 Lines 7.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007.php php 156 Lines 18.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Chart.php php 930 Lines 56.5 KB Sep 30, 2015, 5:27:36 AM

Copyright 2021 Micro Focus or one of its affiliates.

exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Comments.php php 105 Lines 8.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/ContentTypes.php php 95 Lines 10.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/DocProps.php php 110 Lines 8.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Drawing.php php 252 Lines 19.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Rels.php php 129 Lines 13.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/RelsRibbon.php php 19 Lines 2.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/RelsVBA.php php 16 Lines 2.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/StringTable.php php 128 Lines 10.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Style.php php 294 Lines 24.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Theme.php php 387 Lines 24.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Workbook.php php 155 Lines 15.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/Worksheet.php php 542 Lines 43.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel2007/WriterPart.php php 9 Lines 2.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5.php php 413 Lines 33 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5/BIFFwriter.php php 51 Lines 7.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5/Escher.php php 255 Lines 13.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5/Font.php php 47 Lines 3.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5/Parser.php php 666 Lines 55.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5/Workbook.php php 550 Lines 45.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5/Worksheet.php php 1,674 Lines 121.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Excel5/Xf.php php 127 Lines 16.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/Exception.php php 5 Lines 1.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/HTML.php php 532 Lines 45.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/IWriter.php php 1 Lines 1.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument.php php 50 Lines 6.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/Cell/Comment.php php 13 Lines 2.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/Content.php php 147 Lines 12.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/Meta.php php 39 Lines 4.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/MetaInf.php php 39 Lines 4.2 KB Sep 30, 2015,

Copyright 2021 Micro Focus or one of its affiliates.

5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/Mimetype.php php 3 Lines 1.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/Settings.php php 28 Lines 3.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/Styles.php php 44 Lines 5.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/Thumbnails.php php 3 Lines 1.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/OpenDocument/WriterPart.php php 1 Lines 1.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/PDF.php php 19 Lines 3.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/PDF/Core.php php 80 Lines 13.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/PDF/DomPDF.php php 38 Lines 4.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/PDF/mPDF.php php 46 Lines 4.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Classes/PHPExcel/Writer/PDF/tcPDF.php php 47 Lines 5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/01pharSimple.php php 49 Lines 4.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/01simple-download-ods.php php 36 Lines 3.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/01simple-download-pdf.php php 37 Lines 3.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/01simple-download-xls.php php 36 Lines 3.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/01simple-download-xlsx.php php 36 Lines 3.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/01simple.php php 52 Lines 4.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/01simplePCLZip.php php 44 Lines 3.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/02types-xls.php php 100 Lines 7.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/02types.php php 99 Lines 7.6 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/03formulas.php php 72 Lines 6.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/04printing.php php 56 Lines 5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/05featuredemo.inc.php php 226 Lines 18.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/05featuredemo.php php 28 Lines 2.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/06largescale-with-cellcaching-sqlite.php php 57 Lines 4.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/06largescale-with-cellcaching-sqlite3.php php 57 Lines 4.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/06largescale-with-cellcaching.php php 57 Lines 4.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/06largescale-xls.php php 53 Lines 4.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/06largescale.php php 53 Lines 4.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/07reader.php php 27 Lines 2.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/07readerPCLZip.php php 28 Lines 2.9 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/08conditionalformatting.php php 98 Lines 8.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/08conditionalformatting2.php php 69 Lines 5.2 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/09pagebreaks.php php 62 Lines 5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/10autofilter-selection-1.php php 120 Lines 8.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/10autofilter-selection-2.php php 119 Lines 8.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/10autofilter-selection-display.php php 108 Lines 7.4 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/10autofilter.php php 105 Lines 7.5 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/11documentsecurity-xls.php php 47 Lines 4.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/11documentsecurity.php php 47 Lines 4.1 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/12cellProtection.php php 44 Lines 3.7 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/13calculation.php php 140 Lines 10.8 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/14excel5.php php 19 Lines 2.3 KB Sep 30, 2015, 5:27:36 AM
exportold/vendor/phpoffice/phpexcel/Examples/15datavalidation-xls.php php 81 Lines 6 KB Sep 30, 2015, 5:27:37 AM

Copyright 2021 Micro Focus or one of its affiliates.

exportold/vendor/phpoffice/phpexcel/Examples/15datavalidation.php php 81 Lines 6.1 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/16csv.php php 53 Lines 4.6 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/17html.php php 21 Lines 2.4 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/18extendedcalculation.php php 54 Lines 4.3 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/19namedrange.php php 61 Lines 5.2 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/20readexcel5.php php 28 Lines 2.9 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/21pdf.php php 31 Lines 3.3 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/22heavilyformatted.php php 49 Lines 4.2 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/23sharedstyles.php php 56 Lines 4.5 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/24readfilter.php php 26 Lines 2.5 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/25inmemoryimage.php php 36 Lines 3.3 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/26utf8.php php 47 Lines 4.8 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/27imagesexcel5.php php 20 Lines 2.4 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/28iterator.php php 22 Lines 2.3 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/29advancedvaluebinder.php php 96 Lines 7.5 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/30template.php php 31 Lines 3.1 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/31docproperties_write-xls.php php 57 Lines 4.9 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/31docproperties_write.php php 57 Lines 5 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/32chartreadwrite.php php 63 Lines 4.4 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/33chartcreate-area.php php 47 Lines 4.5 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/33chartcreate-bar-stacked.php php 48 Lines 4.7 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/33chartcreate-bar.php php 48 Lines 4.7 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/33chartcreate-column-2.php php 57 Lines 5.2 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/33chartcreate-column.php php 48 Lines 4.7 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/33chartcreate-composite.php php 67 Lines 6.3 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/33chartcreate-line.php php 47 Lines 4.5 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/33chartcreate-multiple-charts.php php 71 Lines 7 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/33chartcreate-pie.php php 65 Lines 6.4 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/33chartcreate-radar.php php 55 Lines 4.9 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/33chartcreate-scatter.php php 48 Lines 4.5 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/33chartcreate-stock.php php 55 Lines 5.1 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/34chartupdate.php php 29 Lines 2.6 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/35chartrender.php php 54 Lines 4.1 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/36chartreadwriteHTML.php php 69 Lines 4.9 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/36chartreadwritePDF.php php 75 Lines 5.6 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/37page_layout_view.php php 33 Lines 3.2 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/38cloneWorksheet.php php 53 Lines 4.1 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/39dropdown.php php 99 Lines 5.8 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/40duplicateStyle.php php 32 Lines 1.7 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/41password.php php 31 Lines 3.1 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/42richText.php php 65 Lines 5.2 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/Excel2003XMLReader.php php 17 Lines 2.3 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/Excel2003XMLTest.xml xml 937 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/GnumericReader.php php 17 Lines 2.3 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/OOCalcReader.php php 17 Lines 2.2 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/OOCalcReaderPCLZip.php php 18 Lines 2.3 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/Quadratic.php php 16 Lines 2.3 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/Quadratic2.php php 13 Lines 2.4 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/SylkReader.php php 11 Lines 1.9 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/XMLReader.php php 15 Lines 2.3 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/Examples/XMLTest.xml xml 20.3 KB Sep 30, 2015, 5:27:37 AM

Copyright 2021 Micro Focus or one of its affiliates.

exportold/vendor/phpoffice/phpexcel/Examples/runall.php php 91 Lines 3.7 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/composer.json json Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/AutoloaderTest.php php 18 Lines 1.5 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/DateTimeTest.php php 169 Lines 15 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/EngineeringTest.php php 236 Lines 19.6 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/FinancialTest.php php 161 Lines 13.7 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/FunctionsTest.php php 87 Lines 7.1 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/LogicalTest.php php 35 Lines 2.7 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/LookupRefTest.php php 16 Lines 1.3 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/MathTrigTest.php php 201 Lines 14.9 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Calculation/TextDataTest.php php 117 Lines 9.7 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/CalculationTest.php php 13 Lines 1.4 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Cell/AdvancedValueBinderTest.php php 45 Lines 2.8 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Cell/DataTypeTest.php php 8 Lines Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Cell/DefaultValueBinderTest.php php 38 Lines 2.2 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Cell/HyperlinkTest.php php 40 Lines 2.3 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/CellTest.php php 94 Lines 8.3 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Chart/DataSeriesValuesTest.php php 20 Lines 1.3 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Chart/LayoutTest.php php 13 Lines Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Chart/LegendTest.php php 62 Lines 3.3 KB Jan 25, 2021, 11:01:50 PM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Reader/XEEValidatorTest.php php 19 Lines 1.4 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/ReferenceHelperTest.php php 35 Lines 1.3 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Shared/CodePageTest.php php 18 Lines 1.5 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Shared/DateTest.php php 67 Lines 5.2 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Shared/FileTest.php php 15 Lines Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Shared/FontTest.php php 32 Lines 2.5 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Shared/PasswordHasherTest.php php 10 Lines Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Shared/StringTest.php php 30 Lines 2.4 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Shared/TimeZoneTest.php php 16 Lines Sep 30, 2015, 5:27:37

Copyright 2021 Micro Focus or one of its affiliates.

AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Style/ColorTest.php php 25 Lines 2 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Style/NumberFormatTest.php php 12 Lines Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/AutoFilter/Column/RuleTest.php php 41 Lines 3.6 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/AutoFilter/ColumnTest.php php 60 Lines 5.2 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/AutoFilterTest.php php 128 Lines 10.4 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/CellCollectionTest.php php 14 Lines Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/ColumnCellIteratorTest.php php 39 Lines 2.7 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/ColumnIteratorTest.php php 40 Lines 2.8 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/RowCellIteratorTest.php php 40 Lines 2.8 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/RowIteratorTest.php php 39 Lines 2.5 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/WorksheetColumnTest.php php 21 Lines 1.4 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/Classes/PHPExcel/Worksheet/WorksheetRowTest.php php 21 Lines 1.4 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/bootstrap.php php 18 Lines 1.4 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/custom/Complex.php php 54 Lines 3.6 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/custom/complexAssert.php php 34 Lines 2 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/phpunit-cc.xml xml 1.5 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/phpunit.xml xml 1 KB Sep 30, 2015, 5:27:37 AM
exportold/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestInvalidUTF-16.xml xml Sep 30, 2015, 5:27:38 AM
exportold/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestInvalidUTF-16BE.xml xml Sep 30, 2015, 5:27:38 AM
exportold/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestInvalidUTF-16LE.xml xml Sep 30, 2015, 5:27:38 AM
exportold/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestInvalidUTF-8.xml xml Sep 30, 2015, 5:27:38 AM
exportold/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestValidUTF-16.xml xml Sep 30, 2015, 5:27:38 AM
exportold/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestValidUTF-16BE.xml xml Sep 30, 2015, 5:27:38 AM
exportold/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestValidUTF-16LE.xml xml Sep 30, 2015, 5:27:38 AM
exportold/vendor/phpoffice/phpexcel/unitTests/rawTestData/Reader/XEETestValidUTF-8.xml xml Sep 30, 2015, 5:27:38 AM
exportold/vendor/phpoffice/phpexcel/unitTests/testDataFileIterator.php php 57 Lines 3.8 KB Sep 30, 2015, 5:27:38 AM
filter-all1.php php 86 Lines 9.7 KB Jan 20, 2019, 8:23:44 PM
filter-all1seach.php php 71 Lines 6.5 KB Aug 2, 2018, 5:46:27 AM
filter-all2.php php 6 Lines 5.7 KB Oct 29, 2018, 10:14:43 PM
filter-all3.php php 39 Lines 5.8 KB Mar 6, 2023, 3:47:52 AM
filter-all4.php php 37 Lines 5.7 KB Mar 3, 2023, 2:56:22 AM
filter-allold3.php php 37 Lines 11.4 KB Jan 26, 2021, 12:34:30 AM
filter-table/filterTable.v1.0.html html 27 Lines 6.7 KB Aug 23, 2014, 1:55:38 AM
filter-table/filterTable.v1.0.src.js typescript 28 Lines 5.8 KB Aug 23, 2014, 1:55:38 AM
img/redit.php php 887 Lines 86.1 KB Jan 29, 2017, 5:02:36 AM
img/redit1.php php 28 Lines 3 KB Jan 29, 2017, 5:02:36 AM

Copyright 2021 Micro Focus or one of its affiliates.

index.php php 35 Lines 2 KB Jun 18, 2019, 10:27:16 PM
jquery-1.12.2.min2.js typescript 5 Lines 95 KB Jan 29, 2017, 5:02:36 AM
jquery.form.js typescript 526 Lines 42.9 KB Feb 1, 2019, 2:04:25 AM
js.js typescript 6 Lines Aug 7, 2018, 4:41:35 AM
mail.php php 14 Lines 3.4 KB Jan 14, 2019, 11:31:37 PM
mail/index.html html 2.4 KB Oct 9, 2017, 3:35:44 AM
mail/mail.php php 19 Lines 2 KB Oct 9, 2017, 3:35:02 AM
mail/phpmailer/PHPMailerAutoload.php php 9 Lines 1.6 KB Nov 22, 2016, 12:10:14 PM
mail/phpmailer/README.md generic 12.2 KB Nov 22, 2016, 12:10:14 PM
mail/phpmailer/class.phpmailer.php php 1,331 Lines 138.5 KB Nov 22, 2016, 12:10:14 PM
mail/phpmailer/class.phpmaileroauth.php php 67 Lines 7 KB Nov 22, 2016, 12:10:14 PM
mail/phpmailer/class.phpmaileroauthgoogle.php php 16 Lines 2.4 KB Nov 22, 2016, 12:10:14 PM
mail/phpmailer/class.pop3.php php 97 Lines 10.7 KB Nov 22, 2016, 12:10:14 PM
mail/phpmailer/class.smtp.php php 362 Lines 39.4 KB Nov 22, 2016, 12:10:14 PM
mail/phpmailer/get_oauth_token.php php 40 Lines 4.9 KB Nov 22, 2016, 12:10:14 PM
mail/thank-you.html html 1.3 KB Oct 9, 2017, 12:36:04 AM
mail2.php php 28 Lines 2.3 KB Aug 27, 2018, 3:42:00 AM
msg-result.php php 1 Lines Jul 20, 2017, 9:18:44 PM
msg.php php 5 Lines Aug 16, 2018, 10:42:32 PM
msg01.php php 5 Lines Jul 18, 2018, 1:07:53 AM
msg15.php php 5 Lines Jul 18, 2018, 1:07:51 AM
options.html html 1 Lines 13.8 KB Jun 8, 2018, 1:43:58 AM
pass.php php 78 Lines 7.4 KB Jan 22, 2019, 3:50:48 AM
pass1.php php 15 Lines 1.3 KB Aug 2, 2018, 5:44:49 AM
pass2.php php 15 Lines 1.3 KB Aug 2, 2018, 5:44:40 AM
pass3.php php 15 Lines 1.3 KB Aug 2, 2018, 5:44:34 AM
pass4.php php 15 Lines 1.3 KB Aug 2, 2018, 5:44:22 AM
pass5-2.php php 23 Lines 1.9 KB Aug 2, 2018, 5:44:10 AM
pass5.php php 15 Lines 1.3 KB Aug 2, 2018, 5:44:15 AM
redit-all.php php 517 Lines 49 KB Aug 2, 2018, 5:42:20 AM
redit-all1.php php   Jul 20, 2017, 9:19:02 PM
redit1.php php 55 Lines 6.8 KB Jan 16, 2020, 2:37:06 AM
redit101.php php 17 Lines 12.2 KB Mar 5, 2023, 10:16:15 PM
redit1011.php php 98 Lines 21.5 KB Jan 10, 2019, 1:55:01 AM
redit10111.php php 38 Lines 22.1 KB Feb 19, 2020, 9:03:08 PM
redit1fil.php php 36 Lines 6 KB Mar 6, 2023, 1:59:55 AM
redit1old.php php 55 Lines 6.8 KB Jan 14, 2019, 2:23:16 AM
redit1p.php php 55 Lines 6.6 KB Oct 29, 2018, 4:56:19 AM
reditf1.php php 49 Lines 5.3 KB Aug 2, 2018, 5:41:47 AM
reditf2.php php 55 Lines 6.8 KB Aug 2, 2018, 5:41:37 AM
reditf3.php php 55 Lines 6.8 KB Aug 2, 2018, 5:41:20 AM
reditf4.php php 53 Lines 6.7 KB Aug 2, 2018, 5:40:41 AM
result.php php 13 Lines 4.6 KB Oct 29, 2018, 4:48:04 AM
script.js typescript 55 Lines 2.5 KB Sep 19, 2018, 10:50:04 PM
selector/index.php php 29 Lines 3.6 KB Jul 17, 2019, 12:46:08 AM
selector/library/Database.php php 177 Lines 10.2 KB Jul 17, 2019, 12:46:08 AM
selector/library/config.php php 20 Lines 1 KB Jul 17, 2019, 12:52:54 AM
selector/magicsuggest/magicsuggest-min.js typescript 1 Lines 21.9 KB Jul 17, 2019, 12:46:08 AM
selector/materialize/README.md generic 2.4 KB Jul 17, 2019, 12:46:08 AM
selector/materialize/css/materialize.css generic 192 KB Jul 17, 2019, 12:46:08 AM
selector/materialize/css/materialize.min.css generic 146.9 KB Jul 17, 2019, 12:46:08 AM

Copyright 2021 Micro Focus or one of its affiliates.

selector/materialize/font/material-design-icons/LICENSE.txt generic 19.7 KB Jul 17, 2019, 12:46:08 AM
selector/materialize/js/materialize.js typescript 2,802 Lines 261.7 KB Jul 17, 2019, 12:46:09 AM
selector/save.php php 7 Lines Jul 17, 2019, 12:46:08 AM
selector/selector.php php 7 Lines 1.1 KB Jul 17, 2019, 12:46:08 AM
send - Copy.php php 31 Lines 120.6 KB Mar 23, 2021, 3:36:42 AM
send.php php 31 Lines 134.1 KB Mar 6, 2023, 4:13:49 AM
send1 - Copy (2).php php 31 Lines 122.8 KB Mar 23, 2021, 8:55:16 PM
send1 - Copy.php php 31 Lines 8.8 KB Mar 23, 2021, 3:10:18 AM
send1.php php 31 Lines 122.7 KB Mar 24, 2021, 12:26:20 AM
send2.php php 1 Lines 1.8 KB Mar 24, 2021, 12:37:44 AM
send222.php php 1 Lines 16.2 KB Mar 23, 2021, 10:43:44 PM
send333.php php 31 Lines 364.5 KB Apr 8, 2021, 8:03:05 PM
showresult.php php 71 Lines 6.4 KB Aug 2, 2018, 11:32:25 PM
test/Searchread1.php php 881 Lines 88.4 KB Jan 29, 2017, 5:02:38 AM
test/in.php php 19 Lines 2.1 KB Jan 29, 2017, 5:02:36 AM
test/msg.php php 5 Lines Jan 29, 2017, 5:02:36 AM
test/selectotchet1.php php 5 Lines Jan 29, 2017, 5:02:38 AM
test/selectotchet2.php php 5 Lines 1 KB Jan 29, 2017, 5:02:38 AM
test/selectotchetfin1.php php 5 Lines 1.2 KB Jan 29, 2017, 5:02:38 AM
test/selectotchettex1.php php 5 Lines 6.8 KB Jan 29, 2017, 5:02:38 AM
test/selectvariant1.php php 5 Lines Jan 29, 2017, 5:02:38 AM
test/sozdanieotchet1.php php 10 Lines 211.2 KB Jan 29, 2017, 5:02:38 AM
upload.php php 8 Lines 2.2 KB Mar 6, 2023, 1:55:34 AM
upload1.php php 5 Lines 1.9 KB Mar 6, 2023, 1:54:26 AM
zsend.php php 31 Lines 7.5 KB Mar 3, 2023, 4:33:01 AM
.html html 5 KB Oct 29, 2018, 7:54:01 PM
_files/analytics.js. generic 42.2 KB Oct 29, 2018, 7:47:07 PM
.php php 20.8 KB Aug 14, 2018, 3:37:20 AM

Reference Elements

Classpath:

No classpath specified during translation

Libdirs:

No libdirs specified during translation

Rulepacks

Valid Rulepacks:

Name: Fortify Secure Coding Rules, Community, Cloud
Version: 2022.4.0.0009
ID: 686C4B2F-0321-4025-B9F4-6E26094B4746
SKU: RUL13242

Name: Fortify Secure Coding Rules, Community, PHP
Version: 2022.4.0.0009
ID: 0f56d3e1-999d-468d-adfa-99c250d3266b
SKU: RUL13244

Copyright 2021 Micro Focus or one of its affiliates.

Name: Fortify Secure Coding Rules, Community, Universal
Version: 2022.4.0.0009
ID: 97b8b0e6-618b-47cf-a7fb-8636faea6b75
SKU: RUL13240

Name: Fortify Secure Coding Rules, Core Cloud
Version: 2022.4.0.0009
ID: BDACC98E-569C-4ECC-92AA-8DD890DF1287
SKU: RUL13049

Name: Fortify Secure Coding Rules, Core, JavaScript
Version: 2022.4.0.0009
ID: BD292C4E-4216-4DB8-96C7-9B607BFD9584
SKU: RUL13059

Name: Fortify Secure Coding Rules, Core, PHP
Version: 2022.4.0.0009
ID: 343CBB32-087C-4A4E-8BD8-273B5F876069
SKU: RUL13058

Name: Fortify Secure Coding Rules, Core, SQL
Version: 2022.4.0.0009
ID: 6494160B-E1DB-41F5-9840-2B1609EE7649
SKU: RUL13004

Name: Fortify Secure Coding Rules, Core, Universal
Version: 2022.4.0.0009
ID: 88D39959-D322-499A-87F3-BC9E1193B07A
SKU: RUL13241

Name: Fortify Secure Coding Rules, Extended, Configuration
Version: 2022.4.0.0009
ID: CD6959FC-0C37-45BE-9637-BAA43C3A4D56
SKU: RUL13005

Name: Fortify Secure Coding Rules, Extended, Content
Version: 2022.4.0.0009
ID: 9C48678C-09B6-474D-B86D-97EE94D38F17
SKU: RUL13067

Name: Fortify Secure Coding Rules, Extended, JavaScript
Version: 2022.4.0.0009
ID: C4D1969E-B734-47D3-87D4-73962C1D32E2
SKU: RUL13141

Name: Fortify Secure Coding Rules, Extended, SQL
Version: 2022.4.0.0009
ID: 4BC5B2FA-C209-4DBC-9C3E-1D3EEFAF135A
SKU: RUL13025

Copyright 2021 Micro Focus or one of its affiliates.

External Metadata:
Version: 2022.4.0.0009

Name: CWE
ID: 3ADB9EE4-5761-4289-8BD3-CBFCC593EBBC
The Common Weakness Enumeration (CWE), co-sponsored and maintained by MITRE, is international in scope and free for public use. CWE provides a unified, measurable set of software weaknesses that is enabling more effective discussion, description, selection, and use of software security tools and services that can find these weaknesses in source code and operational systems as well as better understanding and management of software weaknesses related to architecture and design.

Name: CWE Top 25 2019
ID: 7AF935C9-15AA-45B2-8EEC-0EAE4194ACDE
The 2019 CWE Top 25 Most Dangerous Software Errors lists the most widespread and critical weaknesses that can lead to serious vulnerabilities in software (as demonstrated by the National Vulnerability Database). These weaknesses occur frequently, are often easy to find, and easy to exploit. They are dangerous because they will frequently enable attackers to completely take over the software, steal data, or prevent the software from working at all. The list is the result of heuristic formula that the CWE Team used with a data-driven approach that leveraged the Common Vulnerabilities and Exposure (CVE), National Vulnerability Database (NVD), and Common Vulnerability Scoring System (CVSS). Due to the hierarchical nature of the CWE taxonomy, Fortify considers all CWE IDs which are children of a Top 25 entry, as included within the context of the entry due to the "CHILD-OF" relationship within the hierarchy. Exercise caution if using only this Top 25 list to prioritize auditing efforts because the software under analysis might not align with the assumptions of the heuristic used to define the Top 25. For example, many of these weaknesses are related to C-like languages and the software under analysis might not be within the C-family of languages - thus, many CWEs would not be in scope.

Name: CWE Top 25 2020
ID: E4C1DC51-45BD-469E-BA5D-BABF690F09F4
The 2020 CWE Top 25 Most Dangerous Software Errors lists the most widespread and critical weaknesses that can lead to serious vulnerabilities in software (as demonstrated by the National Vulnerability Database). These weaknesses occur frequently, are often easy to find, and easy to exploit. They are dangerous because they will frequently enable attackers to completely take over the software, steal data, or prevent the software from working at all. The list is the result of heuristic formula that the CWE Team used with a data-driven approach that leveraged the Common Vulnerabilities and Exposure (CVE), National Vulnerability Database (NVD), and Common Vulnerability Scoring System (CVSS). Due to the hierarchical nature of the CWE taxonomy, Fortify considers all CWE IDs which are children of a Top 25 entry, as included within the context of the entry due to the "CHILD-OF" relationship within the hierarchy. Exercise caution if using only this Top 25 list to prioritize auditing efforts because the software under analysis might not align with the assumptions of the heuristic used to define the Top 25. For example, many of these weaknesses are related to C-like languages and the software under analysis might not be within the C-family of languages - thus, many CWEs would not be in scope.

Name: CWE Top 25 2021
ID: FDA85EBD-56E5-4698-86FD-DD52E2F8F32B
The 2021 CWE Top 25 Most Dangerous Software Errors lists the most widespread and critical weaknesses that can lead to serious vulnerabilities in software (as demonstrated by the National Vulnerability Database). These weaknesses occur frequently, are often easy to find, and easy to exploit. They are dangerous because they will frequently enable attackers to completely take over the software, steal data, or prevent the software from working at all. The list is the result of heuristic formula that the CWE Team used with a data-driven approach that leveraged the Common Vulnerabilities and Exposure (CVE), National Vulnerability Database (NVD), and Common Vulnerability Scoring System (CVSS). Due to the hierarchical nature of the CWE taxonomy, Fortify considers all CWE IDs which are children of a Top 25 entry, as included within the context of the entry due to the "CHILD-OF" relationship within the hierarchy. Exercise caution if using only this Top 25 list to prioritize auditing efforts because the software under analysis might not align with the assumptions of the heuristic used to define the Top 25. For example, many of these weaknesses are related to C-like languages and the software under analysis might not be within the C-family of languages - thus, many CWEs would not be in scope.

Copyright 2021 Micro Focus or one of its affiliates.

Name: CWE Top 25 2022
ID: D16E16F3-91AE-4AA3-A943-FCDE765446E5
The 2022 CWE Top 25 Most Dangerous Software Errors lists the most widespread and critical weaknesses that can lead to serious vulnerabilities in software (as demonstrated by the National Vulnerability Database). These weaknesses occur frequently, are often easy to find, and easy to exploit. They are dangerous because they will frequently enable attackers to completely take over the software, steal data, or prevent the software from working at all. The list is the result of heuristic formula that the CWE Team used with a data-driven approach that leveraged the Common Vulnerabilities and Exposure (CVE), National Vulnerability Database (NVD), and Common Vulnerability Scoring System (CVSS). Due to the hierarchical nature of the CWE taxonomy, Fortify considers all CWE IDs which are children of a Top 25 entry, as included within the context of the entry due to the "CHILD-OF" relationship within the hierarchy. Exercise caution if using only this Top 25 list to prioritize auditing efforts because the software under analysis might not align with the assumptions of the heuristic used to define the Top 25. For example, many of these weaknesses are related to C-like languages and the software under analysis might not be within the C-family of languages - thus, many CWEs would not be in scope.

Name: DISA CCI 2
ID: 7F037130-41E5-40F0-B653-7819A4B3E241
The purpose of a Defense Information Systems Agency (DISA) Control Correlation Identifier (CCI) is to provide a standard identifier for policy based requirements which connect high-level policy expressions and low-level technical implementations. Associated with each CCI is a description for each of the singular, actionable, statements compromising an information assurance (IA) control or IA best practice. Using CCI allows high-level policy framework security requirements to be decomposed and explicitly associated with low-level implementations, thus enabling the assessment of related compliance assessment results spanning heterogeneous technologies. The current IA controls and best practices associated with each CCI, that are specified in NIST SP 800-53 Revision 4, can be viewed using the DISA STIG Viewer.
The following table summarizes the number of issues identified across the different CCIs broken down by Fortify Priority Order. The status of a CCI is considered "In Place" when there are no issues reported for a given CCI.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, CCI-003187 is not considered "In Place". Similarly, if the project is missing a Micro Focus Fortify WebInspect scan, or the scan contains any critical findings, CCI-000366 and CCI-000256 are not considered "In Place".

Name: FISMA
ID: B40F9EE0-3824-4879-B9FE-7A789C89307C
The Federal Information Processing Standard (FIPS) 200 document is part of the official series of publications, issued by the National Institute of Standards and Technology (NIST), relating to standards and guidelines adopted and promulgated under the provisions of the Federal Information Security Management Act (FISMA). Specifically, FIPS Publication 200 specifies the "Minimum Security Requirements for Federal Information and Information Systems."

Name: GDPR
ID: 771C470C-9274-4580-8556-C12F5E4BEC51
The EU General Data Protection Regulation (GDPR) replaces the Data Protection Directive 95/46/EC and was designed to harmonize data privacy laws across Europe, to protect and empower all EU citizens data privacy and to reshape the way organizations across the region approach data privacy. Going into effect on May 25, 2018, GDPR provides a framework for organizations on how to handle personal data. According to GDPR regulation personal data "means any information relating to an identified or identifiable natural person ('data subject'); an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person." GDPR articles that pertain to application security and require businesses to protect personal data during design and development of its product and services are:

- Article 25, Data protection by design and by default - which requires "The controller shall implement appropriate technical and

Copyright 2021 Micro Focus or one of its affiliates.

organisational measures for ensuring that, by default, only personal data which are necessary for each specific purpose of the processing are processed."
- Article 32, Security of processing - which requires businesses to protect its systems and applications "from accidental or unlawful destruction, loss, alteration, unauthorized disclosure of, or access to personal data". This report may be used by organizations as a framework to help identify and protect personal data as it relates to application security.

Name: MISRA C 2012
ID: 555A3A66-A0E1-47AF-910C-3F19A6FB2506
Now in its third edition, the Motor Industry Software Reliability Association (MISRA) C Guidelines describe a subset of the C programming language in which there is reduced risk of introducing mistakes in critical systems. While the MISRA C Guidelines focus upon safety-related software development, a subset of the rules reflects security properties. Fortify interprets the MISRA C Guidelines under the context of security and provides correlation of security vulnerability categories to the rules defined by MISRA. Fortify provides these security focused detection mechanisms with the standard rulepacks, however, further support of the MISRA C Guidelines related to safety can be added through the use of custom rules. The results in this report can assist in the creation of a compliance matrix for MISRA.

Name: MISRA C++ 2008
ID: 5D4B75A1-FC91-4B4B-BD4D-C81BBE9604FA
The Motor Industry Software Reliability Association (MISRA) C++ Guidelines describe a subset of the C++ programming language in which there is reduced risk of introducing mistakes in critical systems. While the MISRA C++ Guidelines focus upon safety-related software development, a subset of the rules reflects security properties. Fortify interprets the MISRA C++ Guidelines under the context of security and provides correlation of security vulnerability categories to the rules defined by MISRA. Fortify provides these security focused detection mechanisms with the standard rulepacks, however, further support of the MISRA C++ Guidelines related to safety can be added through the use of custom rules. The results in this report can assist in the creation of a compliance matrix for MISRA.

Name: NIST SP 800-53 Rev.4
ID: 1114583B-EA24-45BE-B7F8-B61201BACDD0
NIST Special Publication 800-53 Revision 4 provides a list of security and privacy controls designed to protect federal organizations and information systems from security threats. The following table summarizes the number of issues identified across the different controls and broken down by Fortify Priority Order.

Name: NIST SP 800-53 Rev.5
ID: 32434089-54F3-49F8-93F8-688B6B2FE8ED
NIST Special Publication 800-53 Revision 5 provides a list of security and privacy controls designed to protect federal organizations and information systems from security threats. The following table summarizes the number of issues identified across the different controls and broken down by Fortify Priority Order.

Name: OWASP ASVS 4.0
ID: 28083E33-760F-4A1A-AADA-738CC60082AD
The OWASP Application Security Verification Standard establishes a framework of security requirements and controls that focus on functional and non-functional security controls for the software development lifecycle based upon a community-driven effort. OWASP ASVS identifies several application security verification levels, with each level increasing depth:

<LI>ASVS Level 1 (L1): for low assurance levels and is completely penetration testable.</LI>
<LI>ASVS Level 2 (L2): for applications that contain sensitive data, which requires protection, and is the recommended level for most apps.</LI>
<LI>ASVS Level 3 (L3): for the most critical applications - applications that perform high value transactions, contain sensitive medical data, or any application that requires the highest level of trust.</LI> </UL>

Copyright 2021 Micro Focus or one of its affiliates.

Name: OWASP Mobile 2014
ID: EEE3F9E7-28D6-4456-8761-3DA56C36F4EE
The OWASP Mobile Top 10 Risks 2014 provides a powerful awareness document for mobile application security. The OWASP Mobile Top 10 represents a broad consensus about what the most critical mobile application security flaws are. Project members include a variety of security experts from around the world who have shared their expertise to produce this list.

Name: OWASP Top 10 2004
ID: 771C470C-9274-4580-8556-C023E4D3ADB4
The OWASP Top Ten 2004 provides a powerful awareness document for web application security. The OWASP Top Ten represents a broad consensus about what the most critical web application security flaws are. Project members include a variety of security experts from around the world who have shared their expertise to produce this list.

Name: OWASP Top 10 2007
ID: 1EB1EC0E-74E6-49A0-BCE5-E6603802987A
The OWASP Top Ten 2007 provides a powerful awareness document for web application security. The OWASP Top Ten represents a broad consensus about what the most critical web application security flaws are. Project members include a variety of security experts from around the world who have shared their expertise to produce this list.

Name: OWASP Top 10 2010
ID: FDCECA5E-C2A8-4BE8-BB26-76A8ECD0ED59
The OWASP Top Ten 2010 provides a powerful awareness document for web application security. The OWASP Top Ten represents a broad consensus about what the most critical web application security flaws are. Project members include a variety of security experts from around the world who have shared their expertise to produce this list.

Name: OWASP Top 10 2013
ID: 1A2B4C7E-93B0-4502-878A-9BE40D2A25C4
The OWASP Top Ten 2013 provides a powerful awareness document for web application security. The OWASP Top Ten represents a broad consensus about what the most critical web application security flaws are. Project members include a variety of security experts from around the world who have shared their expertise to produce this list.

Name: OWASP Top 10 2017
ID: 3C6ECB67-BBD9-4259-A8DB-B49328927248
The OWASP Top Ten 2017 provides a powerful awareness document for web application security focused on informing the community about the consequences of the most common and most important web application security weaknesses. The OWASP Top Ten represents a broad agreement about what the most critical web application security flaws are with consensus being drawn from data collection and survey results. Project members include a variety of security experts from around the world who have shared their expertise to produce this list.

Name: OWASP Top 10 2021
ID: 1887A283-3C0D-453C-AD10-0B451EAF096D0
The OWASP Top 10 2021 provides a powerful awareness document for web application security focused on informing the community about the consequences of the most common and most important web application security weaknesses. The OWASP Top 10 represents a broad agreement about what the most critical web application security flaws are with consensus drawn from data collection and survey results. Project members include a variety of security experts from around the world who have shared their expertise to produce this list.

Name: PCI 1.1
ID: CBDB9D4D-FC20-4C04-AD58-575901CAB531
The Payment Card Industry (PCI) Data Security Standard (DSS) 1.1 compliance standard describes 12 requirements which are organized into 6 logically related groups, which are "control objectives". PCI DSS requirements are applicable if Primary Account

Copyright 2021 Micro Focus or one of its affiliates.

Number (PAN) is stored, processed, or transmitted by the system.

Name: PCI 1.2
ID: 57940BDB-99F0-48BF-BF2E-CFC42BA035E5
Payment Card Industry Data Security Standard Version 1.2 description

Name: PCI 2.0
ID: 8970556D-7F9F-4EA7-8033-9DF39D68FF3E
The PCI DSS 2.0 compliance standard, particularly sections 6.3, 6.5, and 6.6, references the OWASP Top 10 vulnerability categories as the core categories that must be tested for and remediated. The following table summarizes the number of issues identified across the different PCI DSS requirements and broken down by Fortify Priority Order.

Name: PCI 3.0
ID: E2FB0D38-0192-4F03-8E01-FE2A12680CA3
The following is a summary of the application security portions of Payment Card Industry (PCI) Data Security Standard (DSS) v3.0. Fortify tests for 32 application security related requirements across sections 1, 2, 3, 4, 6, 7, 8, and 10 of PCI DSS and reports whether each requirement is In Place or Not In Place to indicate whether requirements are satisfied or not. This report is intended to measure the level of adherence the specific application(s) possess when compared to PCI DSS 3.0 compliance and is not intended to serve as a comprehensive Report on Compliance (ROC). The information contained in this report is targeted at project managers, security auditors, and compliance auditors.

Name: PCI 3.1
ID: AC0D18CF-C1DA-47CF-9F1A-E8EC0A4A717E
The following is a summary of the application security portions of Payment Card Industry (PCI) Data Security Standard (DSS) v3.1. Fortify tests for 31 application security related requirements across sections 1, 2, 3, 4, 6, 7, 8, and 10 of PCI DSS and reports whether each requirement is In Place or Not In Place to indicate whether requirements are satisfied or not. This report is intended to measure the level of adherence the specific application(s) possess when compared to PCI DSS 3.1 compliance and is not intended to serve as a comprehensive Report on Compliance (ROC). The information contained in this report is targeted at project managers, security auditors, and compliance auditors.

Name: PCI 3.2
ID: 4E8431F9-1BA1-41A8-BDBD-087D5826751A
The following is a summary of the application security portions of Payment Card Industry (PCI) Data Security Standard (DSS) v3.2. Fortify tests for 31 application security related requirements across sections 1, 2, 3, 4, 6, 7, 8, and 10 of PCI DSS and reports whether each requirement is In Place or Not In Place to indicate whether requirements are satisfied or not. This report is intended to measure the level of adherence the specific application(s) possess when compared to PCI DSS 3.2 compliance and is not intended to serve as a comprehensive Report on Compliance (ROC). The information contained in this report is targeted at project managers, security auditors, and compliance auditors.

Name: PCI 3.2.1
ID: EADE255F-6561-4EFE-AD31-2914F6BFA329
The following is a summary of the application security portions of Payment Card Industry (PCI) Data Security Standard (DSS) v3.2.1. Fortify tests for 31 application security related requirements across sections 1, 2, 3, 4, 6, 7, 8, and 10 of PCI DSS and reports whether each requirement is In Place or Not In Place to indicate whether requirements are satisfied or not. This report is intended to measure the level of adherence the specific application(s) possess when compared to PCI DSS 3.2.1 compliance and is not intended to serve as a comprehensive Report on Compliance (ROC). The information contained in this report is targeted at project managers, security auditors, and compliance auditors.

Name: PCI SSF 1.0
ID: 0F551543-AF0E-4334-BEDF-1DDCD5F4BF74
The following is a summary of the application security portions of the Secure Software Requirements and Assessment Procedures

Copyright 2021 Micro Focus or one of its affiliates.

defined in the Payment Card Industry (PCI) Software Security Framework (SSF) v1.0. Fortify tests for 23 application security related control objectives across Control Objective sections 2, 3, 4, 5, 6, 7, 8, and A.2 of PCI SSF and reports whether each control objective is In Place or Not In Place to indicate whether requirements are satisfied or not. This report is intended to measure the level of adherence the specific application(s) possess when compared to PCI SSF 1.0 compliance and is not intended to serve as a comprehensive Report on Compliance (ROC). The information contained in this report is targeted at project managers, security auditors, and compliance auditors.

Name: PCI SSF 1.1
ID: 601EA2F3-5EDC-411C-818C-10DC5B29467D
The following is a summary of the application security portions of the Secure Software Requirements and Assessment Procedures defined in the Payment Card Industry (PCI) Software Security Framework (SSF) v1.1. Fortify tests for 31 application security related control objectives across Control Objective sections 2, 3, 4, 5, 6, 7, 8, A.2, B.2, and B.3 of PCI SSF and reports whether each control objective is In Place or Not In Place to indicate whether requirements are satisfied or not. This report is intended to measure the level of adherence the specific application(s) possess when compared to PCI SSF 1.1 compliance and is not intended to serve as a comprehensive Report on Compliance (ROC). The information contained in this report is targeted at project managers, security auditors, and compliance auditors.

Name: SANS Top 25 2009
ID: 939EF193-507A-44E2-ABB7-C00B2168B6D8
The 2009 CWE/SANS Top 25 Programming Errors lists the most significant programming errors that can lead to serious software vulnerabilities. They occur frequently, are often easy to find, and easy to exploit. They are dangerous because they will frequently allow attackers to completely take over the software, steal data, or prevent the software from working at all. The list is the result of collaboration between the SANS Institute, MITRE, and many top software security experts.

Name: SANS Top 25 2010
ID: 72688795-4F7B-484C-88A6-D4757A6121CA
SANS Top 25 2010 Most Dangerous Software Errors provides an enumeration of the most widespread and critical errors, categorized by Common Weakness Enumeration (CWE) identifiers, that lead to serious vulnerabilities in software (http://cwe.mitre.org/). These software errors are often easy to find and exploit. The inherent danger in these errors is that they can allow an attacker to completely take over the software, steal data, or prevent the software from working at all.

Name: SANS Top 25 2011
ID: 92EB4481-1FD9-4165-8E16-F2DE6CB0BD63
SANS Top 25 2011 Most Dangerous Software Errors provides an enumeration of the most widespread and critical errors, categorized by Common Weakness Enumeration (CWE) identifiers, that lead to serious vulnerabilities in software (http://cwe.mitre.org/). These software errors are often easy to find and exploit. The inherent danger in these errors is that they can allow an attacker to completely take over the software, steal data, or prevent the software from working at all.

Name: STIG 3.1
ID: F2FA57EA-5AAA-4DDE-90A5-480BE65CE7E7
Security Technical Implementation Guide Version 3.1 description

Name: STIG 3.10
ID: 788A87FE-C9F9-4533-9095-0379A9B35B12
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APP<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

Copyright 2021 Micro Focus or one of its affiliates.

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APP5080: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APP5100: CAT II is not considered "In Place".

Name: STIG 3.4
ID: 58E2C21D-C70F-4314-8994-B859E24CF855
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APP<I>ID</I>: CAT <I>SEV</I>]. DISA STIG identifies several severities with respect to vulnerabilities:

<LI>CAT I: allow an attacker immediate access into a machine, allow super user access, or bypass a firewall.</LI>
<LI>CAT II: provide information that have a high potential of giving access to an intruder.</LI>
<LI>CAT III: provide information that potentially could lead to compromise.</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

Name: STIG 3.5
ID: DD18E81F-3507-41FA-9DFA-2A9A15B5479F
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APP<I>ID</I>: CAT <I>SEV</I>]. DISA STIG identifies several severities with respect to vulnerabilities:

<LI>CAT I: allow an attacker immediate access into a machine, allow super user access, or bypass a firewall.</LI>
<LI>CAT II: provide information that have a high potential of giving access to an intruder.</LI>
<LI>CAT III: provide information that potentially could lead to compromise.</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

Name: STIG 3.6
ID: 000CA760-0FED-4374-8AA2-6FA3968A07B1
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APP<I>ID</I>: CAT <I>SEV</I>]. DISA STIG identifies several severities with respect to vulnerabilities:

<LI>CAT I: allow an attacker immediate access into a machine, allow super user access, or bypass a firewall.</LI>
<LI>CAT II: provide information that have a high potential of giving access to an intruder.</LI>
<LI>CAT III: provide information that potentially could lead to compromise.</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APP5080: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APP5100: CAT II is not considered "In Place".

Copyright 2021 Micro Focus or one of its affiliates.

Name: STIG 3.7
ID: E69C07C0-81D8-4B04-9233-F3E74167C3D2
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APP<I>ID</I>: CAT <I>SEV</I>]. DISA STIG identifies several severities with respect to vulnerabilities:

<LI>CAT I: allow an attacker immediate access into a machine, allow super user access, or bypass a firewall.</LI>
<LI>CAT II: provide information that have a high potential of giving access to an intruder.</LI>
<LI>CAT III: provide information that potentially could lead to compromise.</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APP5080: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APP5100: CAT II is not considered "In Place".

Name: STIG 3.9
ID: 1A9D736B-2D4A-49D1-88CA-DF464B40D732
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APP<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APP5080: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APP5100: CAT II is not considered "In Place".

Name: STIG 4.1
ID: 95227C50-A9E4-4C9D-A8AF-FD98ABAE1F3C
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Copyright 2021 Micro Focus or one of its affiliates.

Name: STIG 4.10
ID: EF1FF442-1673-4CF1-B7C4-920F1A96A8150
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.11
ID: D9F6C005-1ED5-4685-8A69-79A87A1A9431
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.2
ID: 672C15F8-8822-4E05-8C9E-1A4BAAA7A373
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden

Copyright 2021 Micro Focus or one of its affiliates.

or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.3
ID: A0B313F0-29BD-430B-9E34-6D10F1178506
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.4
ID: ECEC5CA2-7ACA-4B70-BF44-3248B9C6F4F8
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.5
ID: E6010E0A-7F71-4388-B8B7-EE9A02143474
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority

Copyright 2021 Micro Focus or one of its affiliates.

Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.6
ID: EFB9B012-44D6-456D-B197-03D2FD7C7AD6
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.7
ID: B04A1E01-F1C1-48D3-A827-0F70872182D7
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.8
ID: E6805D9F-D5B5-4192-962C-46828FF68507
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>

Copyright 2021 Micro Focus or one of its affiliates.

<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 4.9
ID: 7B9F7B3B-07FC-4B61-99A1-70E3BB23A6A0
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: STIG 5.1
ID: 1E2530B5-61C5-45D0-B479-79CB82DAFF83
Each requirement or recommendation identified by the Defense Information Systems Agency (DISA) STIG is represented by a STIG identifier (STIGID), which corresponds to a checklist item and a severity code [APSC-DV-<I>ID</I>: CAT <I>SEV</I>]. DISA STIG defines three severities with respect to vulnerabilities where their:

<LI>exploitation leads to direct and immediate loss of Confidentiality, Availability, or Integrity (CAT I).</LI>
<LI>exploitation potentially results in loss of Confidentiality, Availability, or Integrity (CAT II).</LI>
<LI>existence degrades protections against loss of Confidentiality, Availability, or Integrity (CAT III).</LI> </UL>

The following table summarizes the number of issues identified across the different STIGIDs broken down by Fortify Priority Order. The status of a STIGID is considered "In Place" when there are no issues reported for a given STIGID.

If the project is missing a Fortify Static Code Analyzer (SCA) scan, or the scan contains findings that have not been fixed, hidden or suppressed, STIGID APSC-DV-003170: CAT II is not considered "In Place". Similarly, if the project is missing a Fortify WebInspect scan, or the scan contains any critical findings, STIGID APSC-DV-001460: CAT II and STIGID APSC-DV-002930: CAT II are not considered "In Place".

Name: WASC 2.00
ID: 74f8081d-dd49-49da-880f-6830cebe9777
The Web Application Security Consortium (WASC) was created as a cooperative effort to standardize, clarify, and organize the threats to the security of a web site. Version 2.00 of their Threat Classification outlines the attacks and weaknesses that can commonly lead to a website being compromised.

Copyright 2021 Micro Focus or one of its affiliates.

Name: WASC 24 + 2
ID: 9DC61E7F-1A48-4711-BBFD-E9DFF537871F
The Web Application Security Consortium (WASC) was created as a cooperative effort to standardize, clarify, and organize the threats to the security of a web site.

Properties

WinForms.CollectionMutationMonitor.Label=WinFormsDataSource
ast.loading.filter=false
awt.toolkit=sun.awt.windows.WToolkit
com.fortify.AuthenticationKey=C:\Users\Administrator\AppData\Local/Fortify/config/tools
com.fortify.Core=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\Core
com.fortify.InstallRoot=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1
com.fortify.InstallationUserName=Administrator
com.fortify.SCAExecutablePath=C:/Program Files/Fortify/Fortify_SCA_and_Apps_21.2.1/bin/sourceanalyzer.exe
com.fortify.TotalPhysicalMemory=17178845184
com.fortify.VS.RequireASPPrecompilation=true
com.fortify.WorkingDirectory=C:\Users\Administrator\AppData\Local/Fortify
com.fortify.locale=en
com.fortify.log.console=false
com.fortify.sca.AddImpliedMethods=true
com.fortify.sca.AntCompilerClass=com.fortify.dev.ant.SCACompiler
com.fortify.sca.AppendLogFile=true
com.fortify.sca.AspnetTranslator=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\Core/private-bin/sca/dotnet/aspcodegen.exe
com.fortify.sca.BuildID=BD
com.fortify.sca.BuildOptions=-b BD -Dcom.fortify.sca.fileextensions.sql=PLSQL F:\BD
com.fortify.sca.BundleControlflowIssues=true
com.fortify.sca.CollectPerformanceData=true
com.fortify.sca.CustomRulesDir=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\Core\config\customrules
com.fortify.sca.DaemonCompilers=com.fortify.sca.util.compilers.GppCompiler,com.fortify.sca.util.compilers.GccCompiler,com.fortify.sca.util.compilers.AppleGppCompiler,com.fortify.sca.util.compilers.AppleGccCompiler,com.fortify.sca.util.compilers.MicrosoftCompiler,com.fortify.sca.util.compilers.MicrosoftLinker,com.fortify.sca.util.compilers.LdCompiler,com.fortify.sca.util.compilers.ArUtil,com.fortify.sca.util.compilers.SunCCompiler,com.fortify.sca.util.compilers.SunCppCompiler,com.fortify.sca.util.compilers.IntelCompiler,com.fortify.sca.util.compilers.ExternalCppAdapter,com.fortify.sca.util.compilers.ClangCompiler
com.fortify.sca.DeadCodeFilter=true
com.fortify.sca.DeadCodeIgnoreTrivialPredicates=true
com.fortify.sca.DefaultAnalyzers=semantic:dataflow:controlflow:nullptr:configuration:content:structural:buffer
com.fortify.sca.DefaultFileTypes=java,rb,erb,jsp,jspx,jspf,tag,tagx,tld,sql,cfm,php,phtml,ctp,pks,pkh,pkb,xml,config,Config,settings,properties,dll,exe,winmd,cs,vb,asax,ascx,ashx,asmx,aspx,master,Master,xaml,baml,cshtml,vbhtml,razor,inc,asp,vbscript,js,jsx,ini,bas,cls,vbs,frm,ctl,html,htm,xsd,wsdd,xmi,py,cfml,cfc,abap,xhtml,cpx,xcfg,jsff,as,mxml,cbl,cob,cscfg,csdef,wadcfg,wadcfgx,appxmanifest,wsdl,plist,bsp,ABAP,BSP,swift,page,trigger,scala,ts,tsx,conf,json,yaml,yml,go,kt,kts,Dockerfile,dockerfile
com.fortify.sca.DefaultJarsDirs=default_jars
com.fortify.sca.DefaultRulesDir=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\Core\config\rules
com.fortify.sca.DisableCFRules=19EF0414-88CD-4882-82FC-BF3A89865666,4E28CEFE-1B94-4711-BF5A-EDA5D1B3E6BF,A2D33B21-FE55-4C53-86C6-2AB5BF343738,7F4CC818-7525-440B-9C68-02267A80179A,7F80BA1C-82E9-4F2A-BBB4-ADFDFB27B215,E650C773-2BB6-42AA-BC29-370AAF0C53ED
com.fortify.sca.DisableDeadCodeElimination=false
com.fortify.sca.DisableFunctionPointers=false

Copyright 2021 Micro Focus or one of its affiliates.

com.fortify.sca.DisableGlobals=false
com.fortify.sca.DisableInferredConstants=false
com.fortify.sca.DotnetDecompiler=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\Core/private-bin/sca/dotnet/dotnet-decompiler.exe
com.fortify.sca.DotnetTranslator=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\Core/private-bin/sca/dotnet/dotnet-translator.exe
com.fortify.sca.EnableInterproceduralConstantResolution=true
com.fortify.sca.EnableNestedWrappers=true
com.fortify.sca.EnableStructuralMatchCache=true
com.fortify.sca.EnableWrapperDetection=true
com.fortify.sca.FVDLDisableDescriptions=false
com.fortify.sca.FVDLDisableProgramData=false
com.fortify.sca.FVDLDisableSnippets=false
com.fortify.sca.FVDLStylesheet=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\Core/resources/sca/fvdl2html.xsl
com.fortify.sca.GoTranslator=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\Core/private-bin/sca/golang/golang.exe
com.fortify.sca.IndirectCallGraphBuilders=WinFormsAdHocFunctionBuilder,VirtualCGBuilder,J2EEIndirectCGBuilder,JNICGBuilder,StoredProcedureResolver,JavaWSCGBuilder,StrutsCGBuilder,DotNetWSCGBuilder,SqlServerSPResolver,ASPCGBuilder,ScriptedCGBuilder,NewJspCustomTagCGBuilder,DotNetCABCGBuilder,StateInjectionCGBuilder,SqlServerSPResolver2,PHPLambdaResolver,JavaWebCGBuilder
com.fortify.sca.JVMArgs=-XX:+UseParallelGC -XX:SoftRefLRUPolicyMSPerMB=3000 --illegal-access=permit --add-exports=jdk.management/com.sun.management.internal=ALL-UNNAMED --add-exports=jdk.scripting.nashorn/jdk.nashorn.internal.runtime=ALL-UNNAMED --add-exports=java.base/jdk.internal.misc=ALL-UNNAMED --add-opens=java.base/java.nio=ALL-UNNAMED --add-opens=java.base/java.util=ALL-UNNAMED --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-opens=java.base/sun.nio.ch=ALL-UNNAMED --add-opens=java.base/java.lang.reflect=ALL-UNNAMED --add-opens=java.base/java.util.regex=ALL-UNNAMED --add-opens=java.base/java.net=ALL-UNNAMED --add-opens=java.base/javax.crypto=ALL-UNNAMED --add-opens=java.management/sun.management=ALL-UNNAMED -Xmx14744M -Xms400M -Xss24M
com.fortify.sca.JavaSourcepathSearch=true
com.fortify.sca.JdkVersion=1.8
com.fortify.sca.LogFile=C:\Users\Administrator\AppData\Local\Fortify\sca21.2\log\sca
com.fortify.sca.LogFileDir=C:\Users\Administrator\AppData\Local\Fortify\sca21.2\log
com.fortify.sca.LogFileExt=.log
com.fortify.sca.LogFileName=sca.log
com.fortify.sca.LogFileNameNoExt=sca
com.fortify.sca.LogFilePath=C:\Users\Administrator\AppData\Local\Fortify\sca21.2\log\sca.log
com.fortify.sca.LogLevel=INFO
com.fortify.sca.LowSeverityCutoff=1.0
com.fortify.sca.MultithreadedAnalysis=true
com.fortify.sca.NoNestedOutTagOutput=org.apache.taglibs.standard.tag.rt.core.RemoveTag,org.apache.taglibs.standard.tag.rt.core.SetTag
com.fortify.sca.OldVbNetExcludeFileTypes=vb,asax,ascx,ashx,asmx,aspx,xaml,cshtml,vbhtml
com.fortify.sca.PHPVersion=7.4
com.fortify.sca.PID=11168
com.fortify.sca.Phase0HigherOrder.Languages=javascript,typescript
com.fortify.sca.Phase0HigherOrder.Level=1
com.fortify.sca.PrintPerformanceDataAfterScan=false
com.fortify.sca.ProjectRoot=C:\Users\Administrator\AppData\Local/Fortify
com.fortify.sca.ProjectRoot=C:\Users\Administrator\AppData\Local/Fortify
com.fortify.sca.PythonVersion=2
com.fortify.sca.RequireMapKeys=jsp_static

Copyright 2021 Micro Focus or one of its affiliates.

com.fortify.sca.ResultsFile=FortifyBD.fpr
com.fortify.sca.ScaMSBuildTargets=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\Core/private-bin/sca/msbuildplugin/Fortify.targets
com.fortify.sca.SolverTimeout=15
com.fortify.sca.SqlLanguage=TSQL
com.fortify.sca.SuppressLowSeverity=true
com.fortify.sca.ThreadCount.NameTableLoading=1
com.fortify.sca.TypeInferenceFunctionTimeout=60
com.fortify.sca.TypeInferenceLanguages=javascript,typescript,python,ruby
com.fortify.sca.TypeInferencePhase0Timeout=300
com.fortify.sca.UnicodeInputFile=true
com.fortify.sca.UniversalBlacklist=.*yyparse.*
com.fortify.sca.alias.mode.csharp=fs
com.fortify.sca.alias.mode.javascript=fi
com.fortify.sca.alias.mode.scala=fi
com.fortify.sca.alias.mode.swift=fs
com.fortify.sca.alias.mode.typescript=fi
com.fortify.sca.alias.mode.vb=fs
com.fortify.sca.analyzer.controlflow.EnableLivenessOptimization=false
com.fortify.sca.analyzer.controlflow.EnableMachineFiltering=false
com.fortify.sca.analyzer.controlflow.EnableRefRuleOptimization=false
com.fortify.sca.analyzer.controlflow.EnableTimeOut=true
com.fortify.sca.compilers.ant=com.fortify.sca.util.compilers.AntAdapter
com.fortify.sca.compilers.ar=com.fortify.sca.util.compilers.ArUtil
com.fortify.sca.compilers.armcc=com.fortify.sca.util.compilers.ArmCcCompiler
com.fortify.sca.compilers.armcpp=com.fortify.sca.util.compilers.ArmCppCompiler
com.fortify.sca.compilers.c++=com.fortify.sca.util.compilers.GppCompiler
com.fortify.sca.compilers.c89=com.fortify.sca.util.compilers.C89Compiler
com.fortify.sca.compilers.cc=com.fortify.sca.util.compilers.GccCompiler
com.fortify.sca.compilers.cl=com.fortify.sca.util.compilers.MicrosoftCompiler
com.fortify.sca.compilers.clearmake=com.fortify.sca.util.compilers.TouchlessCompiler
com.fortify.sca.compilers.devenv=com.fortify.sca.util.compilers.DevenvAdapter
com.fortify.sca.compilers.fortify=com.fortify.sca.util.compilers.FortifyCompiler
com.fortify.sca.compilers.g++=com.fortify.sca.util.compilers.GppCompiler
com.fortify.sca.compilers.g++-*=com.fortify.sca.util.compilers.GppCompiler
com.fortify.sca.compilers.g++2*=com.fortify.sca.util.compilers.GppCompiler
com.fortify.sca.compilers.g++3*=com.fortify.sca.util.compilers.GppCompiler
com.fortify.sca.compilers.g++4*=com.fortify.sca.util.compilers.GppCompiler
com.fortify.sca.compilers.gcc=com.fortify.sca.util.compilers.GccCompiler
com.fortify.sca.compilers.gcc-*=com.fortify.sca.util.compilers.GccCompiler
com.fortify.sca.compilers.gcc2*=com.fortify.sca.util.compilers.GccCompiler
com.fortify.sca.compilers.gcc3*=com.fortify.sca.util.compilers.GccCompiler
com.fortify.sca.compilers.gcc4*=com.fortify.sca.util.compilers.GccCompiler
com.fortify.sca.compilers.gmake=com.fortify.sca.util.compilers.TouchlessCompiler
com.fortify.sca.compilers.gradle=com.fortify.sca.util.compilers.GradleAdapter
com.fortify.sca.compilers.gradlew=com.fortify.sca.util.compilers.GradleAdapter
com.fortify.sca.compilers.icc=com.fortify.sca.util.compilers.IntelCompiler
com.fortify.sca.compilers.icl=com.fortify.sca.util.compilers.MicrosoftCompiler
com.fortify.sca.compilers.icpc=com.fortify.sca.util.compilers.IntelCompiler
com.fortify.sca.compilers.jam=com.fortify.sca.util.compilers.TouchlessCompiler
com.fortify.sca.compilers.javac=com.fortify.sca.util.compilers.JavacCompiler

Copyright 2021 Micro Focus or one of its affiliates.

com.fortify.sca.compilers.ld=com.fortify.sca.util.compilers.LdCompiler
com.fortify.sca.compilers.link=com.fortify.sca.util.compilers.MicrosoftLinker
com.fortify.sca.compilers.make=com.fortify.sca.util.compilers.TouchlessCompiler
com.fortify.sca.compilers.msbuild=com.fortify.sca.util.compilers.MSBuildAdapter
com.fortify.sca.compilers.msdev=com.fortify.sca.util.compilers.MSDevAdapter
com.fortify.sca.compilers.mvn=com.fortify.sca.util.compilers.MavenAdapter
com.fortify.sca.compilers.nmake=com.fortify.sca.util.compilers.TouchlessCompiler
com.fortify.sca.compilers.scalac=com.fortify.sca.util.compilers.ScalacCompiler
com.fortify.sca.compilers.tcc=com.fortify.sca.util.compilers.ArmCcCompiler
com.fortify.sca.compilers.tcpp=com.fortify.sca.util.compilers.ArmCppCompiler
com.fortify.sca.compilers.touchless=com.fortify.sca.util.compilers.FortifyCompiler
com.fortify.sca.compilers.xilink=com.fortify.sca.util.compilers.MicrosoftLinker
com.fortify.sca.cpfe.441.command=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\Core/private-bin/sca/cpfe441.rfct
com.fortify.sca.cpfe.command=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\Core/private-bin/sca/cpfe48.exe
com.fortify.sca.cpfe.file.option=--gen_c_file_name
com.fortify.sca.cpfe.options=--remove_unneeded_entities --suppress_vtbl -tused
com.fortify.sca.cpfe.options=--remove_unneeded_entities --suppress_vtbl -tused
com.fortify.sca.env.exesearchpath=F:\BD;C:\Program Files\Common Files\Oracle\Java\javapath;C:\Program Files (x86)\Common Files\Oracle\Java\javapath;C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\bin;C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\Windows\System32\OpenSSH\;%JAVA_HOME%\bin\;C:\Program Files\dotnet\;C:\Program Files\Microsoft SQL Server\130\Tools\Binn\;C:\Program Files\Microsoft SQL Server\Client SDK\ODBC\170\Tools\Binn\;C:\gradle-7.4.2\bin;C:\Program Files\Java\jdk1.8.0_321\bin;C:\Program Files\Java\jdk1.8.0_333\bin;C:\Users\Administrator\AppData\Local\Microsoft\WindowsApps;C:\Users\Administrator\.dotnet\tools;C:\gradle-7.4.1-all\gradle-7.4.1\bin;C:\Program Files\Java\jdk1.8.0_333\bin;
com.fortify.sca.fileextensions.ABAP=ABAP
com.fortify.sca.fileextensions.BSP=ABAP
com.fortify.sca.fileextensions.Config=XML
com.fortify.sca.fileextensions.Dockerfile=DOCKERFILE
com.fortify.sca.fileextensions.Master=ASPNET
com.fortify.sca.fileextensions.abap=ABAP
com.fortify.sca.fileextensions.appxmanifest=XML
com.fortify.sca.fileextensions.as=ACTIONSCRIPT
com.fortify.sca.fileextensions.asax=ASPNET
com.fortify.sca.fileextensions.ascx=ASPNET
com.fortify.sca.fileextensions.ashx=ASPNET
com.fortify.sca.fileextensions.asmx=ASPNET
com.fortify.sca.fileextensions.asp=ASP
com.fortify.sca.fileextensions.aspx=ASPNET
com.fortify.sca.fileextensions.axml=ASPNET
com.fortify.sca.fileextensions.baml=MSIL
com.fortify.sca.fileextensions.bas=VB6
com.fortify.sca.fileextensions.bsp=ABAP
com.fortify.sca.fileextensions.cbl=COBOL
com.fortify.sca.fileextensions.cfc=CFML
com.fortify.sca.fileextensions.cfm=CFML
com.fortify.sca.fileextensions.cfml=CFML
com.fortify.sca.fileextensions.cls=VB6
com.fortify.sca.fileextensions.cob=COBOL
com.fortify.sca.fileextensions.conf=HOCON
com.fortify.sca.fileextensions.config=XML

Copyright 2021 Micro Focus or one of its affiliates.

com.fortify.sca.fileextensions.cpx=XML
com.fortify.sca.fileextensions.cs=CSHARP
com.fortify.sca.fileextensions.cscfg=XML
com.fortify.sca.fileextensions.csdef=XML
com.fortify.sca.fileextensions.cshtml=ASPNET
com.fortify.sca.fileextensions.ctl=VB6
com.fortify.sca.fileextensions.ctp=PHP
com.fortify.sca.fileextensions.dll=MSIL
com.fortify.sca.fileextensions.dockerfile=DOCKERFILE
com.fortify.sca.fileextensions.erb=RUBY_ERB
com.fortify.sca.fileextensions.exe=MSIL
com.fortify.sca.fileextensions.faces=JSPX
com.fortify.sca.fileextensions.frm=VB6
com.fortify.sca.fileextensions.go=GO
com.fortify.sca.fileextensions.htm=HTML
com.fortify.sca.fileextensions.html=HTML
com.fortify.sca.fileextensions.ini=JAVA_PROPERTIES
com.fortify.sca.fileextensions.java=JAVA
com.fortify.sca.fileextensions.js=TYPESCRIPT
com.fortify.sca.fileextensions.jsff=JSPX
com.fortify.sca.fileextensions.json=JSON
com.fortify.sca.fileextensions.jsp=JSP
com.fortify.sca.fileextensions.jspf=JSP
com.fortify.sca.fileextensions.jspx=JSPX
com.fortify.sca.fileextensions.jsx=TYPESCRIPT
com.fortify.sca.fileextensions.kt=KOTLIN
com.fortify.sca.fileextensions.kts=KOTLIN
com.fortify.sca.fileextensions.master=ASPNET
com.fortify.sca.fileextensions.mdl=MSIL
com.fortify.sca.fileextensions.mod=MSIL
com.fortify.sca.fileextensions.mxml=MXML
com.fortify.sca.fileextensions.page=VISUAL_FORCE
com.fortify.sca.fileextensions.php=PHP
com.fortify.sca.fileextensions.phtml=PHP
com.fortify.sca.fileextensions.pkb=PLSQL
com.fortify.sca.fileextensions.pkh=PLSQL
com.fortify.sca.fileextensions.pks=PLSQL
com.fortify.sca.fileextensions.plist=XML
com.fortify.sca.fileextensions.properties=JAVA_PROPERTIES
com.fortify.sca.fileextensions.py=PYTHON
com.fortify.sca.fileextensions.razor=ASPNET
com.fortify.sca.fileextensions.rb=RUBY
com.fortify.sca.fileextensions.scala=SCALA
com.fortify.sca.fileextensions.settings=XML
com.fortify.sca.fileextensions.sql=SQL
com.fortify.sca.fileextensions.swift=SWIFT
com.fortify.sca.fileextensions.tag=JSP
com.fortify.sca.fileextensions.tagx=JSP
com.fortify.sca.fileextensions.tld=TLD
com.fortify.sca.fileextensions.trigger=APEX_TRIGGER
com.fortify.sca.fileextensions.ts=TYPESCRIPT

Copyright 2021 Micro Focus or one of its affiliates.

com.fortify.sca.fileextensions.tsx=TYPESCRIPT
com.fortify.sca.fileextensions.vb=VB
com.fortify.sca.fileextensions.vbhtml=ASPNET
com.fortify.sca.fileextensions.vbs=VBSCRIPT
com.fortify.sca.fileextensions.vbscript=VBSCRIPT
com.fortify.sca.fileextensions.wadcfg=XML
com.fortify.sca.fileextensions.wadcfgx=XML
com.fortify.sca.fileextensions.winmd=MSIL
com.fortify.sca.fileextensions.wsdd=XML
com.fortify.sca.fileextensions.wsdl=XML
com.fortify.sca.fileextensions.xaml=ASPNET
com.fortify.sca.fileextensions.xcfg=XML
com.fortify.sca.fileextensions.xhtml=JSPX
com.fortify.sca.fileextensions.xmi=XML
com.fortify.sca.fileextensions.xml=XML
com.fortify.sca.fileextensions.xsd=XML
com.fortify.sca.fileextensions.yaml=YAML
com.fortify.sca.fileextensions.yml=YAML
com.fortify.sca.jsp.UseNativeParser=true
com.fortify.sca.lim.RequireTrustedSSLCert=true
com.fortify.sca.parser.python.ignore.module.1=test.badsyntax_future3
com.fortify.sca.parser.python.ignore.module.2=test.badsyntax_future4
com.fortify.sca.parser.python.ignore.module.3=test.badsyntax_future5
com.fortify.sca.parser.python.ignore.module.4=test.badsyntax_future6
com.fortify.sca.parser.python.ignore.module.5=test.badsyntax_future7
com.fortify.sca.parser.python.ignore.module.6=test.badsyntax_future8
com.fortify.sca.parser.python.ignore.module.7=test.badsyntax_future9
com.fortify.sca.parser.python.ignore.module.8=test.badsyntax_nocaret
com.fortify.sca.skip.libraries.AngularJS=angular.js,angular.min.js,angular-animate.js,angular-aria.js,angular_1_router.js,angular-cookies.js,angular-message-format.js,angular-messages.js,angular-mocks.js,angular-parse-ext.js,angular-resource.js,angular-route.js,angular-sanitize.js,angular-touch.js
com.fortify.sca.skip.libraries.ES6=es6-shim.min.js,system-polyfills.js,shims_for_IE.js
com.fortify.sca.skip.libraries.jQuery=jquery.js,jquery.min.js,jquery-migrate.js,jquery-migrate.min.js,jquery-ui.js,jquery-ui.min.js,jquery.mobile.js,jquery.mobile.min.js,jquery.color.js,jquery.color.min.js,jquery.color.svg-names.js,jquery.color.svg-names.min.js,jquery.color.plus-names.js,jquery.color.plus-names.min.js,jquery.tools.min.js
com.fortify.sca.skip.libraries.javascript=bootstrap.js,bootstrap.min.js,typescript.js,typescriptServices.js
com.fortify.sca.skip.libraries.typescript=typescript.d.ts,typescriptServices.d.ts
com.fortify.sca.use.ctran=true
com.fortify.search.defaultSyntaxVer=2
com.sun.management.jmxremote=true
dotnet.install.dir=C:\Windows\Microsoft.NET\Framework64\
dotnet.sdk.v11.install.dir=
dotnet.sdk.v20.install.dir=
dotnet.sdk.v3x.install.dir=
dotnet.v30.referenceAssemblies=
dotnet.v35.referenceAssemblies=
file.encoding=Cp1252
file.separator=\
ide.hide.excluded.files=false
idea.home.path=C:\Users\ADMINI~1\AppData\Local\Temp\1\
idea.ignore.disabled.plugins=true

Copyright 2021 Micro Focus or one of its affiliates.

idea.io.use.nio2=true
idea.plugins.compatible.build=201.6668.13
java.awt.graphicsenv=sun.awt.Win32GraphicsEnvironment
java.awt.headless=true
java.awt.printerjob=sun.awt.windows.WPrinterJob
java.class.path=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\Core\lib\exe\sca-exe.jar
java.class.version=55.0
java.home=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\jre
java.io.tmpdir=C:\Users\ADMINI~1\AppData\Local\Temp\1\
java.library.path=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\bin;C:\Windows\Sun\Java\bin;C:\Windows\system32;C:\Windows;C:\Program Files\Common Files\Oracle\Java\javapath;C:\Program Files (x86)\Common Files\Oracle\Java\javapath;C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\bin;C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\Windows\System32\OpenSSH\;%JAVA_HOME%\bin\;C:\Program Files\dotnet\;C:\Program Files\Microsoft SQL Server\130\Tools\Binn\;C:\Program Files\Microsoft SQL Server\Client SDK\ODBC\170\Tools\Binn\;C:\gradle-7.4.2\bin;C:\Program Files\Java\jdk1.8.0_321\bin;C:\Program Files\Java\jdk1.8.0_333\bin;C:\Users\Administrator\AppData\Local\Microsoft\WindowsApps;C:\Users\Administrator\.dotnet\tools;C:\gradle-7.4.1-all\gradle-7.4.1\bin;C:\Program Files\Java\jdk1.8.0_333\bin;;.
java.rmi.server.randomIDs=true
java.runtime.name=OpenJDK Runtime Environment
java.runtime.version=11.0.10+9-LTS
java.specification.name=Java Platform API Specification
java.specification.vendor=Oracle Corporation
java.specification.version=11
java.vendor=Azul Systems, Inc.
java.vendor.url=http://www.azulsystems.com/
java.vendor.url.bug=http://www.azulsystems.com/support/
java.vendor.version=Zulu11.45+27-CA
java.version=11.0.10
java.version.date=2021-01-19
java.vm.compressedOopsMode=Zero based
java.vm.info=mixed mode
java.vm.name=OpenJDK 64-Bit Server VM
java.vm.specification.name=Java Virtual Machine Specification
java.vm.specification.vendor=Oracle Corporation
java.vm.specification.version=11
java.vm.vendor=Azul Systems, Inc.
java.vm.version=11.0.10+9-LTS
jdk.debug=release
jdk.vendor.version=Zulu11.45+27-CA
line.separator=

log4j.configurationFile=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\Core\config\log4j2.xml
log4j.isThreadContextMapInheritable=true
max.file.path.length=255
os.arch=amd64
os.name=Windows Server 2019
os.version=10.0
path.separator=;
project.structure.add.tools.jar.to.new.jdk=false
psi.incremental.reparse.depth.limit=1000

Copyright 2021 Micro Focus or one of its affiliates.

psi.track.invalidation=true
stderr.isatty=true
stdout.isatty=true
sun.arch.data.model=64
sun.boot.library.path=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\jre\bin
sun.cpu.endian=little
sun.cpu.isalist=amd64
sun.desktop=windows
sun.io.unicode.encoding=UnicodeLittle
sun.java.command=sourceanalyzer -Djava.awt.headless=true -Dcom.sun.management.jmxremote=true -XX:+UseParallelGC -XX:SoftRefLRUPolicyMSPerMB=3000 --illegal-access=permit --add-exports=jdk.management/com.sun.management.internal=ALL-UNNAMED --add-exports=jdk.scripting.nashorn/jdk.nashorn.internal.runtime=ALL-UNNAMED --add-exports=java.base/jdk.internal.misc=ALL-UNNAMED --add-opens=java.base/java.nio=ALL-UNNAMED --add-opens=java.base/java.util=ALL-UNNAMED --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-opens=java.base/sun.nio.ch=ALL-UNNAMED --add-opens=java.base/java.lang.reflect=ALL-UNNAMED --add-opens=java.base/java.util.regex=ALL-UNNAMED --add-opens=java.base/java.net=ALL-UNNAMED --add-opens=java.base/javax.crypto=ALL-UNNAMED --add-opens=java.management/sun.management=ALL-UNNAMED -Dwin32.LocalAppdata=C:\Users\Administrator\AppData\Local -Ddotnet.install.dir=C:\Windows\Microsoft.NET\Framework64\ -Ddotnet.sdk.v11.install.dir= -Ddotnet.sdk.v20.install.dir= -Ddotnet.sdk.v3x.install.dir= -Ddotnet.v30.referenceAssemblies= -Ddotnet.v35.referenceAssemblies= -Dcom.fortify.sca.env.exesearchpath=F:\BD;C:\Program Files\Common Files\Oracle\Java\javapath;C:\Program Files (x86)\Common Files\Oracle\Java\javapath;C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\bin;C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\Windows\System32\OpenSSH\;%JAVA_HOME%\bin\;C:\Program Files\dotnet\;C:\Program Files\Microsoft SQL Server\130\Tools\Binn\;C:\Program Files\Microsoft SQL Server\Client SDK\ODBC\170\Tools\Binn\;C:\gradle-7.4.2\bin;C:\Program Files\Java\jdk1.8.0_321\bin;C:\Program Files\Java\jdk1.8.0_333\bin;C:\Users\Administrator\AppData\Local\Microsoft\WindowsApps;C:\Users\Administrator\.dotnet\tools;C:\gradle-7.4.1-all\gradle-7.4.1\bin;C:\Program Files\Java\jdk1.8.0_333\bin; -Dcom.fortify.sca.ProjectRoot=C:\Users\Administrator\AppData\Local/Fortify -Dstdout.isatty=true -Dstderr.isatty=true -Dcom.fortify.sca.PID=11168 -Xmx14744M -Xms400M -Xss24M -Dcom.fortify.TotalPhysicalMemory=17178845184 -Dcom.fortify.sca.JVMArgs=-XX:+UseParallelGC -XX:SoftRefLRUPolicyMSPerMB=3000 --illegal-access=permit --add-exports=jdk.management/com.sun.management.internal=ALL-UNNAMED --add-exports=jdk.scripting.nashorn/jdk.nashorn.internal.runtime=ALL-UNNAMED --add-exports=java.base/jdk.internal.misc=ALL-UNNAMED --add-opens=java.base/java.nio=ALL-UNNAMED --add-opens=java.base/java.util=ALL-UNNAMED --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-opens=java.base/sun.nio.ch=ALL-UNNAMED --add-opens=java.base/java.lang.reflect=ALL-UNNAMED --add-opens=java.base/java.util.regex=ALL-UNNAMED --add-opens=java.base/java.net=ALL-UNNAMED --add-opens=java.base/javax.crypto=ALL-UNNAMED --add-opens=java.management/sun.management=ALL-UNNAMED -Xmx14744M -Xms400M -Xss24M -Djava.class.path=C:\Program Files\Fortify\Fortify_SCA_and_Apps_21.2.1\Core\lib\exe\sca-exe.jar -b BD -Dcom.fortify.sca.Phase0HigherOrder.Languages=javascript,typescript -scan -f FortifyBD.fpr
sun.jnu.encoding=Cp1252
sun.management.compiler=HotSpot 64-Bit Tiered Compilers
sun.os.patch.level=
sun.stderr.encoding=cp437
sun.stdout.encoding=cp437
user.country=US
user.dir=F:\BD
user.home=C:\Users\Administrator
user.language=en
user.name=Administrator
user.script=

Copyright 2021 Micro Focus or one of its affiliates.

user.timezone=America/Los_Angeles
user.variant=
win32.LocalAppdata=C:\Users\Administrator\AppData\Local

Commandline Arguments

-b
BD
-Dcom.fortify.sca.Phase0HigherOrder.Languages=javascript,typescript

-scan
-f
FortifyBD.fpr

Warnings

[12004] The PHP frontend was unable to resolve the following include:
../Build/PHPExcel.phar at F:\BD\export\vendor\phpoffice\phpexcel\Examples\01pharSimple.php:37.
PEAR.php at F:\BD\excel\Classes\PHPExcel\Shared\JAMA\examples\Stats.php:22.
jpgraph_contour.php at F:\BD\excel\Classes\PHPExcel\Chart\Renderer\jpgraph.php:689.
PHPExcel/IOFactory.php at F:\BD\export\vendor\phpoffice\phpexcel\Examples\32chartreadwrite.php:43.
jpgraph_bar.php at F:\BD\excel\Classes\PHPExcel\Chart\Renderer\jpgraph.php:559.
testDataFileIterator.php at
F:\BD\export\vendor\phpoffice\phpexcel\unitTests\Classes\PHPExcel\Calculation\DateTimeTest.php:4.
jpgraph_regstat.php at F:\BD\excel\Classes\PHPExcel\Chart\Renderer\jpgraph.php:571.
jpgraph_pie3d.php at F:\BD\excel\Classes\PHPExcel\Chart\Renderer\jpgraph.php:596.
extras/ntlm_sasl_client.php at F:\BD\mail\phpmailer\class.smtp.php:473.
jpgraph_radar.php at F:\BD\excel\Classes\PHPExcel\Chart\Renderer\jpgraph.php:667.
jpgraph_line.php at F:\BD\excel\Classes\PHPExcel\Chart\Renderer\jpgraph.php:537.
oleread.inc at F:\BD\1\reader.php:31.
jpgraph_scatter.php at F:\BD\excel\Classes\PHPExcel\Chart\Renderer\jpgraph.php:570.
sections/Home.php at F:\BD\excel\Classes\PHPExcel\Shared\JAMA\docs\docs.php:4.
jpgraph_pie.php at F:\BD\excel\Classes\PHPExcel\Chart\Renderer\jpgraph.php:594.
jpgraph_stock.php at F:\BD\excel\Classes\PHPExcel\Chart\Renderer\jpgraph.php:678.
/var/www/vhosts/nas.gov.kg/zaiaviteli-basrmir.nas.gov.kg/1/reader.php at F:\BD\1\price.php:27.
PHPExcel/Calculation.php at F:\BD\export\vendor\phpoffice\phpexcel\Examples\Quadratic2.php:40.
vendor/autoload.php at F:\BD\mail\phpmailer\get_oauth_token.php:18.
custom/complexAssert.php at
F:\BD\export\vendor\phpoffice\phpexcel\unitTests\Classes\PHPExcel\Calculation\EngineeringTest.php:4.
Archive/Tar.php at F:\BD\excel\Classes\PHPExcel\Shared\JAMA\docs\download.php:30.
[12005] It may be helpful to use -php-source-root to resolve relative includes.
[1380] Rule "3A93A824-B310-4416-99DC-31447DA5AB1A", line 15:86: Cannot compare types ConfigConstruct and String with operator "==".
[1380] Rule "7D66460B-4BE7-43C7-89C8-529865D43691", line 21:31: Cannot compare types ConfigConstruct and String with operator "==".
[1380] Rule "ADA73929-D0AF-40C5-AA98-166215551DF8", line 15:86: Cannot compare types ConfigConstruct and String with operator "==".
[1380] Rule "BDB932AC-FE34-4C75-976C-00F52EA446EB", line 15:86: Cannot compare types ConfigConstruct and String with operator "==".
[1380] Rule "D1462DDB-453B-4D00-AFF5-7AAD9899DCF1", line 15:86: Cannot compare types ConfigConstruct and String with operator "==".

Copyright 2021 Micro Focus or one of its affiliates.

[1380] Rule "E8305F25-CAC4-4150-9489-A1251EC9673B", line 36:59: Cannot compare types ConfigConstruct and String with operator "==".
[1380] Rule "F41C974D-6255-414E-8214-159FC5C236E5", line 15:86: Cannot compare types ConfigConstruct and String with operator "==".
[1380] Rule "FD9F2FE3-DC68-4795-B8E8-254A29A8F914", line 15:86: Cannot compare types ConfigConstruct and String with operator "==".
[1380] Rule "0F568D2E-F045-447B-B1D4-A4D6926D0ADD", line 21:31: Cannot compare types ConfigConstruct and String with operator "==".
[1380] Rule "196B72C1-2B2E-4B55-BFBF-179DD1C46190", line 36:59: Cannot compare types ConfigConstruct and String with operator "==".
[1380] Rule "5F85EA15-CDF4-47F2-8EEF-59B7B36F0140", line 15:86: Cannot compare types ConfigConstruct and String with operator "==".
[1380] Rule "82967415-95DC-469D-BBE2-B333E9C1E2A6", line 24:92: Cannot compare types String and Number with operator "==".
[1380] Rule "844D5E05-8DC2-42DE-88C8-62A74AA85BE2", line 15:86: Cannot compare types ConfigConstruct and String with operator "==".
[1380] Rule "A80151CB-14C0-460A-B565-B33F5E535733", line 21:92: Cannot compare types String and Number with operator "==".
[1380] Rule "AECB3F18-7050-4251-AE02-3A5CE23E7B6D", line 15:86: Cannot compare types ConfigConstruct and String with operator "==".
[1380] Rule "C86E13B4-6E94-410E-93F7-FE1781BCF5AD", line 24:92: Cannot compare types String and Number with operator "==".
[1380] Rule "CD06C348-09FB-4B12-81AB-3251F4EB6A9A", line 15:86: Cannot compare types ConfigConstruct and String with operator "==".
[1380] Rule "DB71D699-56FA-4B86-98B2-3506A8D41A8A", line 15:86: Cannot compare types ConfigConstruct and String with operator "==".
[1380] Rule "E4FFE05B-07E9-4DE0-BB19-FE67C92B8CF1", line 15:86: Cannot compare types ConfigConstruct and String with operator "==".
[1380] Rule "E74D92A2-57BC-4EDD-99C5-3DD63D3244EA", line 21:92: Cannot compare types String and Number with operator "==".
[1395] File:F:/BD/chosen/index.html, Message:Misplaced closing tag "</div>" (line: 282, col: 8).

Copyright 2021 Micro Focus or one of its affiliates.

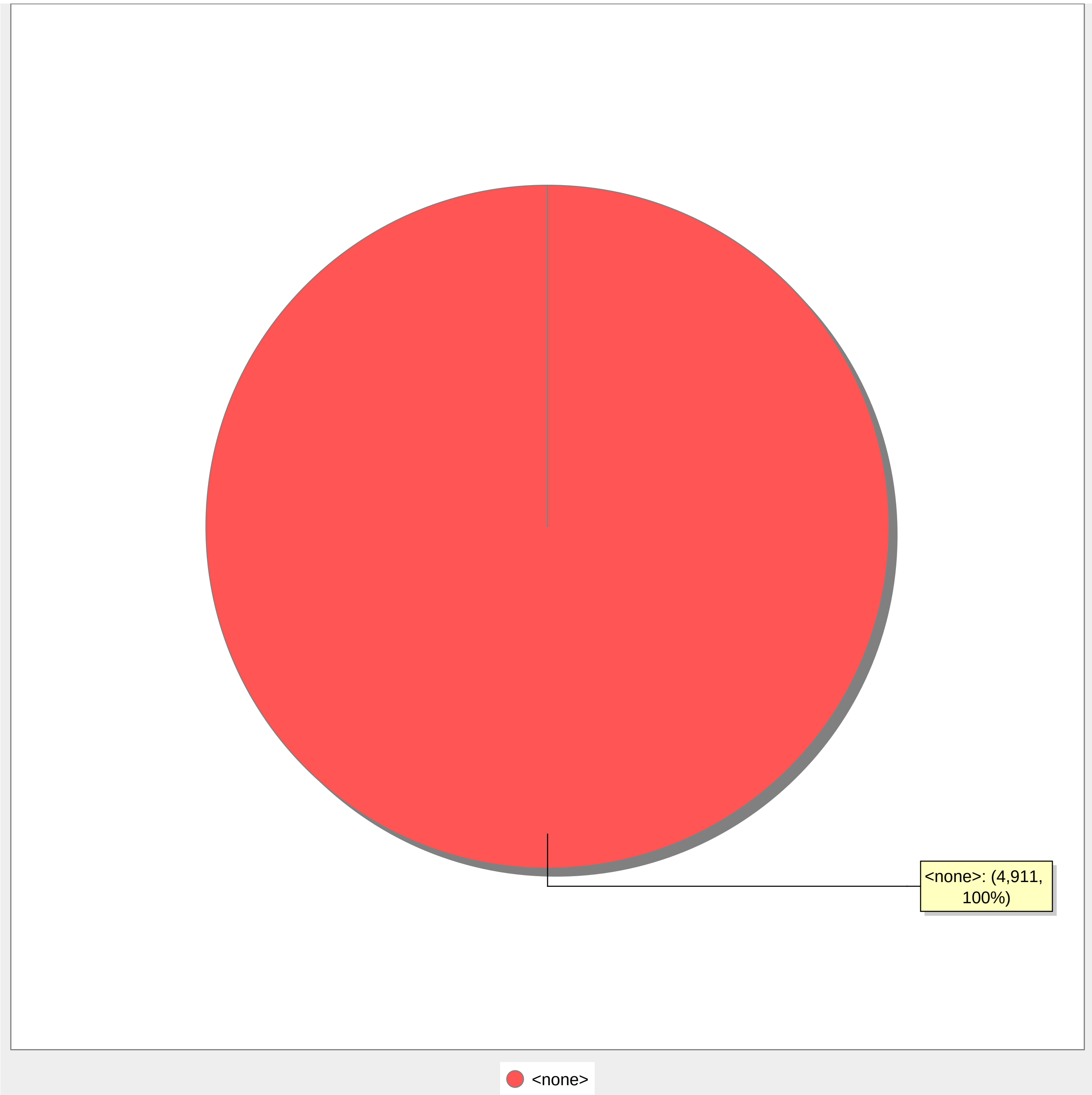
Issue Count by Category

Issues by Category	
Cross-Site Scripting: Persistent	3932
SQL Injection	711
Password Management: Hardcoded Password	111
Cross-Site Scripting: Reflected	63
Header Manipulation: SMTP	27
Privacy Violation	23
Password Management: Empty Password	13
Cross-Site Scripting: DOM	5
Open Redirect	4
Path Manipulation	3
Privacy Violation: Autocomplete	3
System Information Leak: External	3
Credential Management: Hardcoded API Credentials	2
Dynamic Code Evaluation: Code Injection	2
Dynamic Code Evaluation: Insecure Transport	2
Key Management: Hardcoded Encryption Key	2
Privacy Violation: Shoulder Surfing	2
Dangerous Function	1
Key Management: Empty Encryption Key	1
Weak Encryption	1

Copyright 2021 Micro Focus or one of its affiliates.

Issue Breakdown by Analysis

Issues by Analysis



Copyright 2021 Micro Focus or one of its affiliates.