vul_files_90 Scan Report

Project Name	vul_files_90
Scan Start	Thursday, January 9, 2025 3:21:57 PM
Preset	Checkmarx Default
Scan Time	01h:52m:32s
Lines Of Code Scanned	288325
Files Scanned	13
Report Creation Time	Thursday, January 9, 2025 5:11:28 PM
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085
Team	CxServer
Checkmarx Version	8.7.0
Scan Type	Full
Source Origin	LocalPath
Density	7/10000 (Vulnerabilities/LOC)
Visibility	Public

Filter Settings

Severity

Included: High, Medium, Low, Information

Excluded: None

Result State

Included: Confirmed, Not Exploitable, To Verify, Urgent, Proposed Not Exploitable

Excluded: None

Assigned to

Included: All

Categories

Included:

Uncategorized	All
Custom	All
PCI DSS v3.2	All
OWASP Top 10 2013	All
FISMA 2014	All
NIST SP 800-53	All
OWASP Top 10 2017	All
OWASP Mobile Top 10 2016	All

Excluded:

Uncategorized	None
Custom	None
PCI DSS v3.2	None
OWASP Top 10 2013	None
FISMA 2014	None

NIST SP 800-53	None
OWASP Top 10 2017	None
OWASP Mobile Top 10 2016	None

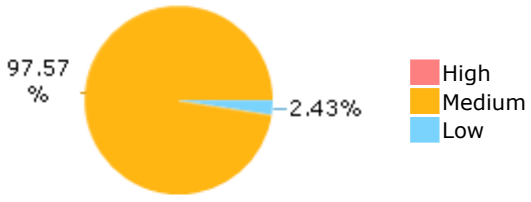
Results Limit

Results limit per query was set to 50

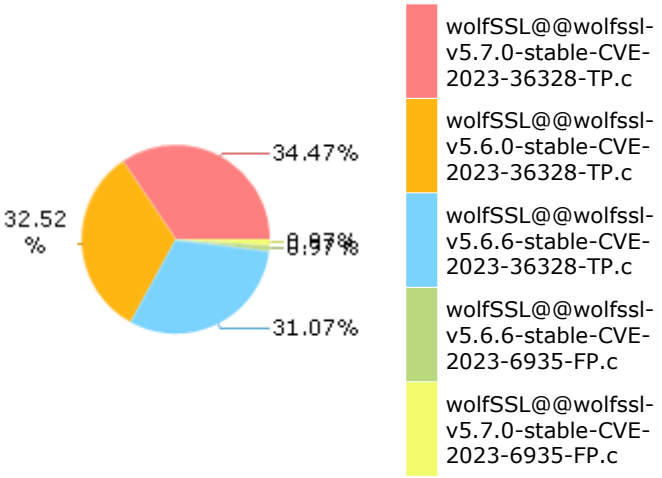
Selected Queries

Selected queries are listed in Result Summary

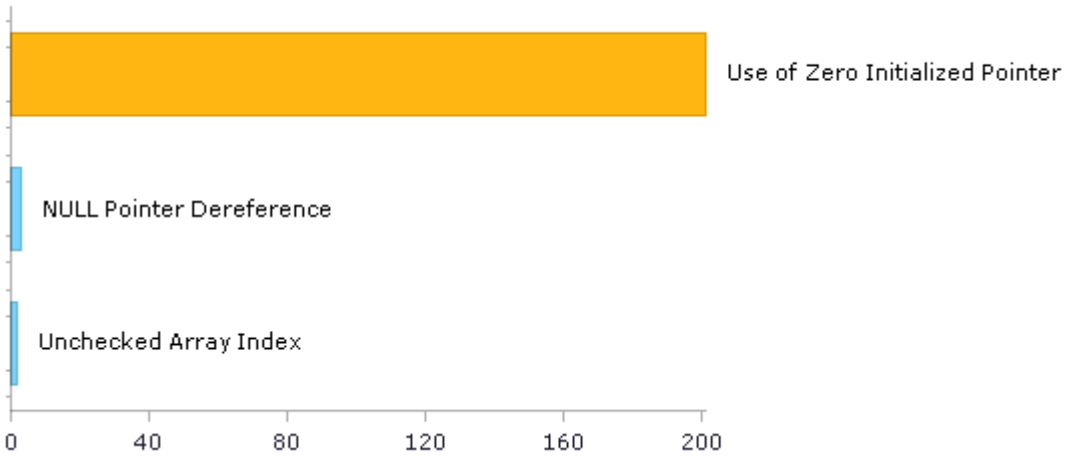
Result Summary



Most Vulnerable Files



Top 5 Vulnerabilities



Scan Summary - OWASP Top 10 2017

Further details and elaboration about vulnerabilities and risks can be found at: OWASP Top 10 2017

Category	Threat Agent	Exploitability	Weakness Prevalence	Weakness Detectability	Technical Impact	Business Impact	Issues Found	Best Fix Locations
A1-Injection	App. Specific	EASY	COMMON	EASY	SEVERE	App. Specific	3	3
A2-Broken Authentication	App. Specific	EASY	COMMON	AVERAGE	SEVERE	App. Specific	0	0
A3-Sensitive Data Exposure	App. Specific	AVERAGE	WIDESPREAD	AVERAGE	SEVERE	App. Specific	0	0
A4-XML External Entities (XXE)	App. Specific	AVERAGE	COMMON	EASY	SEVERE	App. Specific	0	0
A5-Broken Access Control*	App. Specific	AVERAGE	COMMON	AVERAGE	SEVERE	App. Specific	0	0
A6-Security Misconfiguration	App. Specific	EASY	WIDESPREAD	EASY	MODERATE	App. Specific	0	0
A7-Cross-Site Scripting (XSS)	App. Specific	EASY	WIDESPREAD	EASY	MODERATE	App. Specific	0	0
A8-Insecure Deserialization	App. Specific	DIFFICULT	COMMON	AVERAGE	SEVERE	App. Specific	0	0
A9-Using Components with Known Vulnerabilities*	App. Specific	AVERAGE	WIDESPREAD	AVERAGE	MODERATE	App. Specific	0	0
A10-Insufficient Logging & Monitoring	App. Specific	AVERAGE	WIDESPREAD	DIFFICULT	MODERATE	App. Specific	0	0

* Project scan results do not include all relevant queries. Presets and\or Filters should be changed to include all relevant standard queries.

Scan Summary - OWASP Top 10 2013

Further details and elaboration about vulnerabilities and risks can be found at: OWASP Top 10 2013

Category	Threat Agent	Attack Vectors	Weakness Prevalence	Weakness Detectability	Technical Impact	Business Impact	Issues Found	Best Fix Locations
A1-Injection	EXTERNAL, INTERNAL, ADMIN USERS	EASY	COMMON	AVERAGE	SEVERE	ALL DATA	0	0
A2-Broken Authentication and Session Management	EXTERNAL, INTERNAL USERS	AVERAGE	WIDESPREAD	AVERAGE	SEVERE	AFFECTED DATA AND FUNCTIONS	0	0
A3-Cross-Site Scripting (XSS)	EXTERNAL, INTERNAL, ADMIN USERS	AVERAGE	VERY WIDESPREAD	EASY	MODERATE	AFFECTED DATA AND SYSTEM	0	0
A4-Insecure Direct Object References	SYSTEM USERS	EASY	COMMON	EASY	MODERATE	EXPOSED DATA	0	0
A5-Security Misconfiguration	EXTERNAL, INTERNAL, ADMIN USERS	EASY	COMMON	EASY	MODERATE	ALL DATA AND SYSTEM	0	0
A6-Sensitive Data Exposure	EXTERNAL, INTERNAL, ADMIN USERS, USERS BROWSERS	DIFFICULT	UNCOMMON	AVERAGE	SEVERE	EXPOSED DATA	0	0
A7-Missing Function Level Access Control*	EXTERNAL, INTERNAL USERS	EASY	COMMON	AVERAGE	MODERATE	EXPOSED DATA AND FUNCTIONS	0	0
A8-Cross-Site Request Forgery (CSRF)	USERS BROWSERS	AVERAGE	COMMON	EASY	MODERATE	AFFECTED DATA AND FUNCTIONS	0	0
A9-Using Components with Known Vulnerabilities*	EXTERNAL USERS, AUTOMATED TOOLS	AVERAGE	WIDESPREAD	DIFFICULT	MODERATE	AFFECTED DATA AND FUNCTIONS	0	0
A10-Unvalidated Redirects and Forwards	USERS BROWSERS	AVERAGE	WIDESPREAD	DIFFICULT	MODERATE	AFFECTED DATA AND FUNCTIONS	0	0

* Project scan results do not include all relevant queries. Presets and\or Filters should be changed to include all relevant standard queries.

Scan Summary - PCI DSS v3.2

Category	Issues Found	Best Fix Locations
PCI DSS (3.2) - 6.5.1 - Injection flaws - particularly SQL injection	0	0
PCI DSS (3.2) - 6.5.2 - Buffer overflows	0	0
PCI DSS (3.2) - 6.5.3 - Insecure cryptographic storage	0	0
PCI DSS (3.2) - 6.5.4 - Insecure communications	0	0
PCI DSS (3.2) - 6.5.5 - Improper error handling*	0	0
PCI DSS (3.2) - 6.5.7 - Cross-site scripting (XSS)	0	0
PCI DSS (3.2) - 6.5.8 - Improper access control	0	0
PCI DSS (3.2) - 6.5.9 - Cross-site request forgery	0	0
PCI DSS (3.2) - 6.5.10 - Broken authentication and session management	0	0

***** Project scan results do not include all relevant queries. Presets and\or Filters should be changed to include all relevant standard queries.

Scan Summary - FISMA 2014

Category	Description	Issues Found	Best Fix Locations
Access Control	Organizations must limit information system access to authorized users, processes acting on behalf of authorized users, or devices (including other information systems) and to the types of transactions and functions that authorized users are permitted to exercise.	0	0
Audit And Accountability*	Organizations must: (i) create, protect, and retain information system audit records to the extent needed to enable the monitoring, analysis, investigation, and reporting of unlawful, unauthorized, or inappropriate information system activity; and (ii) ensure that the actions of individual information system users can be uniquely traced to those users so they can be held accountable for their actions.	0	0
Configuration Management	Organizations must: (i) establish and maintain baseline configurations and inventories of organizational information systems (including hardware, software, firmware, and documentation) throughout the respective system development life cycles; and (ii) establish and enforce security configuration settings for information technology products employed in organizational information systems.	0	0
Identification And Authentication*	Organizations must identify information system users, processes acting on behalf of users, or devices and authenticate (or verify) the identities of those users, processes, or devices, as a prerequisite to allowing access to organizational information systems.	0	0
Media Protection	Organizations must: (i) protect information system media, both paper and digital; (ii) limit access to information on information system media to authorized users; and (iii) sanitize or destroy information system media before disposal or release for reuse.	0	0
System And Communications Protection	Organizations must: (i) monitor, control, and protect organizational communications (i.e., information transmitted or received by organizational information systems) at the external boundaries and key internal boundaries of the information systems; and (ii) employ architectural designs, software development techniques, and systems engineering principles that promote effective information security within organizational information systems.	0	0
System And Information Integrity	Organizations must: (i) identify, report, and correct information and information system flaws in a timely manner; (ii) provide protection from malicious code at appropriate locations within organizational information systems; and (iii) monitor information system security alerts and advisories and take appropriate actions in response.	0	0

***** Project scan results do not include all relevant queries. Presets and\or Filters should be changed to include all relevant standard queries.

Scan Summary - NIST SP 800-53

Category	Issues Found	Best Fix Locations
AC-12 Session Termination (P2)	0	0
AC-3 Access Enforcement (P1)	0	0
AC-4 Information Flow Enforcement (P1)	0	0
AC-6 Least Privilege (P1)	0	0
AU-9 Protection of Audit Information (P1)	0	0
CM-6 Configuration Settings (P2)	0	0
IA-5 Authenticator Management (P1)	0	0
IA-6 Authenticator Feedback (P2)	0	0
IA-8 Identification and Authentication (Non-Organizational Users) (P1)	0	0
SC-12 Cryptographic Key Establishment and Management (P1)	0	0
SC-13 Cryptographic Protection (P1)	0	0
SC-17 Public Key Infrastructure Certificates (P1)	0	0
SC-18 Mobile Code (P2)	0	0
SC-23 Session Authenticity (P1)*	0	0
SC-28 Protection of Information at Rest (P1)	0	0
SC-4 Information in Shared Resources (P1)	0	0
SC-5 Denial of Service Protection (P1)*	204	39
SC-8 Transmission Confidentiality and Integrity (P1)	0	0
SI-10 Information Input Validation (P1)*	2	2
SI-11 Error Handling (P2)*	0	0
SI-15 Information Output Filtering (P0)	0	0
SI-16 Memory Protection (P1)	0	0

***** Project scan results do not include all relevant queries. Presets and\or Filters should be changed to include all relevant standard queries.

Scan Summary - OWASP Mobile Top 10 2016

Category	Description	Issues Found	Best Fix Locations
M1-Improper Platform Usage	This category covers misuse of a platform feature or failure to use platform security controls. It might include Android intents, platform permissions, misuse of TouchID, the Keychain, or some other security control that is part of the mobile operating system. There are several ways that mobile apps can experience this risk.	0	0
M2-Insecure Data Storage	This category covers insecure data storage and unintended data leakage.	0	0
M3-Insecure Communication	This category covers poor handshaking, incorrect SSL versions, weak negotiation, cleartext communication of sensitive assets, etc.	0	0
M4-Insecure Authentication	This category captures notions of authenticating the end user or bad session management. This can include: -Failing to identify the user at all when that should be required -Failure to maintain the user's identity when it is required -Weaknesses in session management	0	0
M5-Insufficient Cryptography	The code applies cryptography to a sensitive information asset. However, the cryptography is insufficient in some way. Note that anything and everything related to TLS or SSL goes in M3. Also, if the app fails to use cryptography at all when it should, that probably belongs in M2. This category is for issues where cryptography was attempted, but it wasnt done correctly.	0	0
M6-Insecure Authorization	This is a category to capture any failures in authorization (e.g., authorization decisions in the client side, forced browsing, etc.). It is distinct from authentication issues (e.g., device enrolment, user identification, etc.). If the app does not authenticate users at all in a situation where it should (e.g., granting anonymous access to some resource or service when authenticated and authorized access is required), then that is an authentication failure not an authorization failure.	0	0
M7-Client Code Quality	This category is the catch-all for code-level implementation problems in the mobile client. That's distinct from server-side coding mistakes. This would capture things like buffer overflows, format string vulnerabilities, and various other code-level mistakes where the solution is to rewrite some code that's running on the mobile device.	0	0
M8-Code Tampering	This category covers binary patching, local resource modification, method hooking, method swizzling, and dynamic memory modification. Once the application is delivered to the mobile device, the code and data resources are resident there. An attacker can either directly modify the code, change the contents of memory dynamically, change or replace the system APIs that the application uses, or	0	0

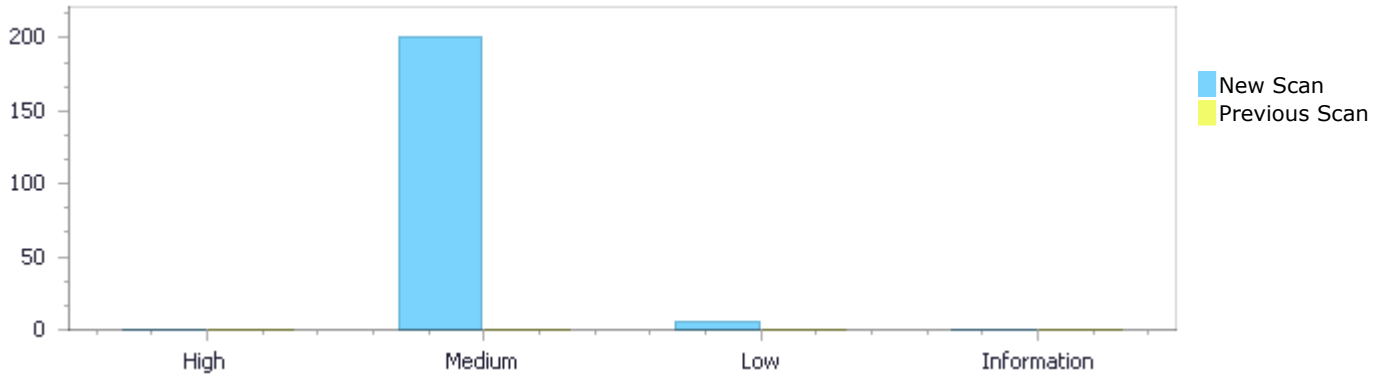
	modify the application's data and resources. This can provide the attacker a direct method of subverting the intended use of the software for personal or monetary gain.		
M9-Reverse Engineering	This category includes analysis of the final core binary to determine its source code, libraries, algorithms, and other assets. Software such as IDA Pro, Hopper, otool, and other binary inspection tools give the attacker insight into the inner workings of the application. This may be used to exploit other nascent vulnerabilities in the application, as well as revealing information about back end servers, cryptographic constants and ciphers, and intellectual property.	0	0
M10-Extraneous Functionality	Often, developers include hidden backdoor functionality or other internal development security controls that are not intended to be released into a production environment. For example, a developer may accidentally include a password as a comment in a hybrid app. Another example includes disabling of 2-factor authentication during testing.	0	0

Scan Summary - Custom

Category	Issues Found	Best Fix Locations
Must audit	0	0
Check	0	0
Optional	0	0

Results Distribution By Status First scan of the project

	High	Medium	Low	Information	Total
New Issues	0	201	5	0	206
Recurrent Issues	0	0	0	0	0
Total	0	201	5	0	206
Fixed Issues	0	0	0	0	0



Results Distribution By State

	High	Medium	Low	Information	Total
Confirmed	0	0	0	0	0
Not Exploitable	0	0	0	0	0
To Verify	0	201	5	0	206
Urgent	0	0	0	0	0
Proposed Not Exploitable	0	0	0	0	0
Total	0	201	5	0	206

Result Summary

Vulnerability Type	Occurrences	Severity
Use of Zero Initialized Pointer	201	Medium
NULL Pointer Dereference	3	Low
Unchecked Array Index	2	Low

10 Most Vulnerable Files

High and Medium Vulnerabilities

File Name	Issues Found
wolfSSL@@wolfssl-v5.7.0-stable-CVE-2023-36328-TP.c	70
wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	66
wolfSSL@@wolfssl-v5.6.6-stable-CVE-2023-36328-TP.c	63
wolfSSL@@wolfssl-v5.6.6-stable-CVE-2023-6935-FP.c	1
wolfSSL@@wolfssl-v5.7.0-stable-CVE-2023-6935-FP.c	1

Scan Results Details

Use of Zero Initialized Pointer

Query Path:
CPP\Cx\CPP Medium Threat\Use of Zero Initialized Pointer Version:1

Categories

NIST SP 800-53: SC-5 Denial of Service Protection (P1)

Description

Use of Zero Initialized Pointer\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=4
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 773.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	829
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.     a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_mul_2d (mp_int * a, int b, mp_int * c)

```
....
829.          c->dp[(c->used)++] = r;
```

Use of Zero Initialized Pointer\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=5
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 773.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	829
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method void mp_free (mp_int * a)

```
....
213.      a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_mul_2d (mp_int * a, int b, mp_int * c)

```
....
829.          c->dp[(c->used)++] = r;
```

Use of Zero Initialized Pointer\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=6
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 1697.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	1760
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.     a->dp = NULL;
```

File Name	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method	int s_mp_add (mp_int * a, mp_int * b, mp_int * c)

```
....
1760.            *tmpc = x->dp[i] + u;
```

Use of Zero Initialized Pointer\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=7
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 1697.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	1760
Object	dp	dp

Code Snippet

File Name	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method	void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method	int s_mp_add (mp_int * a, mp_int * b, mp_int * c)

```
....
1760.            *tmpc = x->dp[i] + u;
```

Use of Zero Initialized Pointer\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=8
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 1379.

	Source	Destination

File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	1400
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.    a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_cmp_mag (mp_int * a, mp_int * b)

```
....
1400.    tmpb = b->dp + (a->used - 1);
```

Use of Zero Initialized Pointer\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=9
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 1379.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	1400
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method void mp_free (mp_int * a)

```
....
213.        a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_cmp_mag (mp_int * a, mp_int * b)

```
....
1400.     tmpb = b->dp + (a->used - 1);
```

Use of Zero Initialized Pointer\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=10
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 1379.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	1397
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_cmp_mag (mp_int * a, mp_int * b)

```
....
1397.     tmpa = a->dp + (a->used - 1);
```

Use of Zero Initialized Pointer\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=11
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 1379.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Line	162	1397
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.    a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_cmp_mag (mp_int * a, mp_int * b)

```
....
1397.    tmpa = a->dp + (a->used - 1);
```

Use of Zero Initialized Pointer\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=12
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 566.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	596
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method void mp_free (mp_int * a)

```
....
213.      a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method void mp_rshb (mp_int *c, int x)

```
....
596.      tmpc = c->dp + (c->used - 1);
```

Use of Zero Initialized Pointer\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=13
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 566.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	596
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_init (mp_int * a)

```
....
162.    a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method void mp_rshb (mp_int *c, int x)

```
....
596.      tmpc = c->dp + (c->used - 1);
```

Use of Zero Initialized Pointer\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=14
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 616.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	640
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method void mp_rshd (mp_int * a, int b)

```
....
640.       top = a->dp + b;
```

Use of Zero Initialized Pointer\Path 12:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=15
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 616.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	640
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.    a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method void mp_rshd (mp_int * a, int b)

```
....
640.       top = a->dp + b;
```

Use of Zero Initialized Pointer\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=16
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 838.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	864
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method void mp_free (mp_int * a)

```
....
213.      a->dp    = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_lshd (mp_int * a, int b)

```
....
864.      bottom = a->dp + a->used - 1 - b;
```

Use of Zero Initialized Pointer\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=17
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 838.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	864
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_init (mp_int * a)

```
....
162.    a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_lshd (mp_int * a, int b)

```
....
864.      bottom = a->dp + a->used - 1 - b;
```

Use of Zero Initialized Pointer\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=18
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 838.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	861
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method void mp_free (mp_int * a)

```
....
213.      a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_lshd (mp_int * a, int b)

```
....
861.      top = a->dp + a->used - 1;
```

Use of Zero Initialized Pointer\Path 16:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=19
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 838.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	861
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.    a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_lshd (mp_int * a, int b)

```
....
861.        top = a->dp + a->used - 1;
```

Use of Zero Initialized Pointer\Path 17:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=20
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 255.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	269
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.    a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_count_bits (const mp_int * a)

```
....
269.     q = a->dp[a->used - 1];
```

Use of Zero Initialized Pointer\Path 18:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=21
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 255.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	269
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_count_bits (const mp_int * a)

```
....
269.     q = a->dp[a->used - 1];
```

Use of Zero Initialized Pointer\Path 19:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=22
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 1599.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	1616
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.    a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_div_2(mp_int * a, mp_int * b)

```
....
1616.       tmpa = a->dp + b->used - 1;
```

Use of Zero Initialized Pointer\Path 20:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=23
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 1599.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	1616
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_div_2(mp_int * a, mp_int * b)

```
....
1616.       tmpa = a->dp + b->used - 1;
```

Use of Zero Initialized Pointer\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=24
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 1599.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	1635
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_init (mp_int * a)

```
....
162.     a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_div_2(mp_int * a, mp_int * b)

```
....
1635.       tmpb = b->dp + b->used;
```

Use of Zero Initialized Pointer\Path 22:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=25
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 1599.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Line	213	1635
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_div_2(mp_int * a, mp_int * b)

```
....
1635.      tmpb = b->dp + b->used;
```

Use of Zero Initialized Pointer\Path 23:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=26
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 1599.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	1619
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.     a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_div_2(mp_int * a, mp_int * b)

```
....
1619.      tmpb = b->dp + b->used - 1;
```

Use of Zero Initialized Pointer\Path 24:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=27
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 1599.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	1619
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method void mp_free (mp_int * a)

```
....
213.      a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_div_2(mp_int * a, mp_int * b)

```
....
1619.      tmpb = b->dp + b->used - 1;
```

Use of Zero Initialized Pointer\Path 25:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=28
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 2904.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	2927
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.    a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_set_bit (mp_int * a, int b)

```
....
2927.       a->dp[i] |= ((mp_digit)1) << (b % DIGIT_BIT);
```

Use of Zero Initialized Pointer\Path 26:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=29
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 2904.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	2927
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_set_bit (mp_int * a, int b)

```
....
2927.       a->dp[i] |= ((mp_digit)1) << (b % DIGIT_BIT);
```

Use of Zero Initialized Pointer\Path 27:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=30
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3316.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	3389
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_init (mp_int * a)

```
....
162.    a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int fast_s_mp_sqr (mp_int * a, mp_int * b)

```
....
3389.           _W += ((mp_word)a->dp[ix>>1])*((mp_word)a->dp[ix>>1]);
```

Use of Zero Initialized Pointer\Path 28:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=31
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3316.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	3389
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int fast_s_mp_sqr (mp_int * a, mp_int * b)

```
....
3389.           _W += ((mp_word)a->dp[ix>>1])*((mp_word)a->dp[ix>>1]);
```

Use of Zero Initialized Pointer\Path 29:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=32
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3316.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	3389
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_init (mp_int * a)

```
....
162.     a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int fast_s_mp_sqr (mp_int * a, mp_int * b)

```
....
3389.           _W += ((mp_word)a->dp[ix>>1])*((mp_word)a->dp[ix>>1]);
```

Use of Zero Initialized Pointer\Path 30:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=33
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3316.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	3389
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int fast_s_mp_sqr (mp_int * a, mp_int * b)

```
....
3389.            _W += ((mp_word)a->dp[ix>>1])*((mp_word)a->dp[ix>>1]);
```

Use of Zero Initialized Pointer\Path 31:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=34
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3316.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	3366
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.     a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int fast_s_mp_sqr (mp_int * a, mp_int * b)

```
....
3366.         tmpy = a->dp + ty;
```

Use of Zero Initialized Pointer\Path 32:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=35
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3316.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	3366
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int fast_s_mp_sqr (mp_int * a, mp_int * b)

```
....
3366.         tmpy = a->dp + ty;
```

Use of Zero Initialized Pointer\Path 33:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=36
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3316.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	3365
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.     a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int fast_s_mp_sqr (mp_int * a, mp_int * b)

```
....
3365.         tmpx = a->dp + tx;
```

Use of Zero Initialized Pointer\Path 34:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=37
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3316.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	3365
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int fast_s_mp_sqr (mp_int * a, mp_int * b)

```
....
3365.         tmpx = a->dp + tx;
```

Use of Zero Initialized Pointer\Path 35:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=38
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3538.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	3573
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_init (mp_int * a)

```
....
162.     a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int s_mp_sqr (mp_int * a, mp_int * b)

```
....
3573.         r       = ((mp_word)tmpx) * ((mp_word)a->dp[iy]);
```

Use of Zero Initialized Pointer\Path 36:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=39
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3538.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Line	213	3573
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int s_mp_sqr (mp_int * a, mp_int * b)

```
....
3573.         r       = ((mp_word)tmpx) * ((mp_word)a->dp[iy]);
```

Use of Zero Initialized Pointer\Path 37:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=40
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3538.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	3566
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.     a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int s_mp_sqr (mp_int * a, mp_int * b)

```
....
3566.       tmpx        = a->dp[ix];
```

Use of Zero Initialized Pointer\Path 38:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=41
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3538.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	3566
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method void mp_free (mp_int * a)

```
....
213.      a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int s_mp_sqr (mp_int * a, mp_int * b)

```
....
3566.      tmpx         = a->dp[ix];
```

Use of Zero Initialized Pointer\Path 39:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=42
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3538.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	3557
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.     a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int s_mp_sqr (mp_int * a, mp_int * b)

```
....
3557.           ((mp_word)a->dp[ix])*((mp_word)a->dp[ix]);
```

Use of Zero Initialized Pointer\Path 40:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=43
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3538.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	3557
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int s_mp_sqr (mp_int * a, mp_int * b)

```
....
3557.           ((mp_word)a->dp[ix])*((mp_word)a->dp[ix]);
```

Use of Zero Initialized Pointer\Path 41:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=44
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3538.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	3557
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.    a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int s_mp_sqr (mp_int * a, mp_int * b)

```
....
3557.           ((mp_word)a->dp[ix])*((mp_word)a->dp[ix]);
```

Use of Zero Initialized Pointer\Path 42:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=45
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3538.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	3557
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int s_mp_sqr (mp_int * a, mp_int * b)

```
....
3557.           ((mp_word)a->dp[ix])*((mp_word)a->dp[ix]);
```

Use of Zero Initialized Pointer\Path 43:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=46
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 2652.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	2699
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_init (mp_int * a)

```
....
162.     a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_montgomery_reduce (mp_int * x, mp_int * n, mp_digit rho)

```
....
2699.          tmpx = x->dp + ix;
```

Use of Zero Initialized Pointer\Path 44:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=47
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 2652.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	2699
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_montgomery_reduce (mp_int * x, mp_int * n, mp_digit rho)

```
....
2699.         tmpx = x->dp + ix;
```

Use of Zero Initialized Pointer\Path 45:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=48
Status New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 2652.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	2687
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.     a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_montgomery_reduce (mp_int * x, mp_int * n, mp_digit rho)

```
....
2687.       mu = (mp_digit) (((mp_word)x->dp[ix]) * ((mp_word)rho) &
MP_MASK);
```

Use of Zero Initialized Pointer\Path 46:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=49
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 2652.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	2687
Object	dp	dp

Code Snippet

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c

Method int mp_montgomery_reduce (mp_int * x, mp_int * n, mp_digit rho)

```
....
2687.       mu = (mp_digit) (((mp_word)x->dp[ix]) * ((mp_word)rho) &
MP_MASK);
```

Use of Zero Initialized Pointer\Path 47:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=50
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3156.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	3207
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.     a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_mul_2(mp_int * a, mp_int * b)

```
....
3207.       tmpb = b->dp + b->used;
```

Use of Zero Initialized Pointer\Path 48:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=51
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 207 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3218.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	213	3236
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method void mp_free (mp_int * a)

```
....
213.       a->dp     = NULL;
```

File Name	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method	int mp_div_3 (mp_int * a, mp_int *c, mp_digit * d)

```
....
3236.       w = (w << ((mp_word)DIGIT_BIT)) | ((mp_word)a->dp[ix]);
```

Use of Zero Initialized Pointer\Path 49:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=52
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 3218.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	3236
Object	dp	dp

Code Snippet

File Name	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method	int mp_init (mp_int * a)

```
....
162.    a->dp = NULL;
```

File Name	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method	int mp_div_3 (mp_int * a, mp_int *c, mp_digit * d)

```
....
3236.       w = (w << ((mp_word)DIGIT_BIT)) | ((mp_word)a->dp[ix]);
```

Use of Zero Initialized Pointer\Path 50:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=53
Status	New

The variable declared in dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 155 is not initialized when it is used by dp at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 668.

	Source	Destination

File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	162	719
Object	dp	dp

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_init (mp_int * a)

```
....
162.    a->dp = NULL;
```

File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_mod_2d (mp_int * a, int b, mp_int * c)

```
....
719.       c->dp[bmax - 1] &=
```

NULL Pointer Dereference

Query Path:
CPP\Cx\CPP Low Visibility\NULL Pointer Dereference Version:1

Categories

NIST SP 800-53: SC-5 Denial of Service Protection (P1)
OWASP Top 10 2017: A1-Injection

Description
NULL Pointer Dereference\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=1
Status	New

The variable declared in 0 at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 1467 is not initialized when it is used by a at wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c in line 1467.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Line	1474	1474
Object	0	a

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.0-stable-CVE-2023-36328-TP.c
Method int mp_set (mp_int * a, mp_digit b)

```
....
1474.       a->used  = (a->dp[0] != 0) ? 1 : 0;
```

NULL Pointer Dereference\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=2
Status	New

The variable declared in 0 at wolfSSL@@wolfssl-v5.6.6-stable-CVE-2023-36328-TP.c in line 1487 is not initialized when it is used by a at wolfSSL@@wolfssl-v5.6.6-stable-CVE-2023-36328-TP.c in line 1487.

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.6-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.6.6-stable-CVE-2023-36328-TP.c
Line	1494	1494
Object	0	a

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.6-stable-CVE-2023-36328-TP.c
Method int mp_set (mp_int * a, mp_digit b)

```
....
1494.       a->used  = (a->dp[0] != 0) ? 1 : 0;
```

NULL Pointer Dereference\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=3
Status	New

The variable declared in 0 at wolfSSL@@wolfssl-v5.7.0-stable-CVE-2023-36328-TP.c in line 1487 is not initialized when it is used by a at wolfSSL@@wolfssl-v5.7.0-stable-CVE-2023-36328-TP.c in line 1487.

	Source	Destination
File	wolfSSL@@wolfssl-v5.7.0-stable-CVE-2023-36328-TP.c	wolfSSL@@wolfssl-v5.7.0-stable-CVE-2023-36328-TP.c
Line	1494	1494
Object	0	a

Code Snippet
File Name wolfSSL@@wolfssl-v5.7.0-stable-CVE-2023-36328-TP.c
Method int mp_set (mp_int * a, mp_digit b)

```
....
1494.       a->used  = (a->dp[0] != 0) ? 1 : 0;
```

Unchecked Array Index

Query Path:
CPP\Cx\CPP Low Visibility\Unchecked Array Index Version:1

Categories

NIST SP 800-53: SI-10 Information Input Validation (P1)

Description

Unchecked Array Index\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=205
Status	New

	Source	Destination
File	wolfSSL@@wolfssl-v5.6.6-stable-CVE-2023-6935-FP.c	wolfSSL@@wolfssl-v5.6.6-stable-CVE-2023-6935-FP.c
Line	3560	3560
Object	i	i

Code Snippet
File Name wolfSSL@@wolfssl-v5.6.6-stable-CVE-2023-6935-FP.c
Method rng : random number generator */

```
....
3560.                           signed char c;
```

Unchecked Array Index\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050096&projectid=50085&pathid=206
Status	New

	Source	Destination
File	wolfSSL@@wolfssl-v5.7.0-stable-CVE-2023-6935-FP.c	wolfSSL@@wolfssl-v5.7.0-stable-CVE-2023-6935-FP.c
Line	3583	3583
Object	i	i

Code Snippet
File Name wolfSSL@@wolfssl-v5.7.0-stable-CVE-2023-6935-FP.c

Method rng : random number generator */

```
....
3583.                           signed char c;
```

Use of Zero Initialized Pointer

Risk

What might happen

A null pointer dereference is likely to cause a run-time exception, a crash, or other unexpected behavior.

Cause

How does it happen

Variables which are declared without being assigned will implicitly retain a null value until they are assigned. The null value can also be explicitly set to a variable, to ensure clear out its contents. Since null is not really a value, it may not have object variables and methods, and any attempt to access contents of a null object, instead of verifying it is set beforehand, will result in a null pointer dereference exception.

General Recommendations

How to avoid it

- For any variable that is created, ensure all logic flows between declaration and use assign a non-null value to the variable first.
- Enforce null checks on any received variable or object before it is dereferenced, to ensure it does not contain a null assigned to it elsewhere.
- Consider the need to assign null values in order to overwrite initialized variables. Consider reassigning or releasing these variables instead.

Source Code Examples

CPP

Explicit NULL Dereference

```
char * input = NULL;
printf("%s", input);
```

Implicit NULL Dereference

```
char * input;
printf("%s", input);
```

```

```

Java
Explicit Null Dereference

```
Object o = null;
out.println(o.getClass());
```

NULL Pointer Dereference

Risk

What might happen

A null pointer dereference is likely to cause a run-time exception, a crash, or other unexpected behavior.

Cause

How does it happen

Variables which are declared without being assigned will implicitly retain a null value until they are assigned. The null value can also be explicitly set to a variable, to ensure clear out its contents. Since null is not really a value, it may not have object variables and methods, and any attempt to access contents of a null object, instead of verifying it is set beforehand, will result in a null pointer dereference exception.

General Recommendations

How to avoid it

- For any variable that is created, ensure all logic flows between declaration and use assign a non-null value to the variable first.
- Enforce null checks on any received variable or object before it is dereferenced, to ensure it does not contain a null assigned to it elsewhere.
- Consider the need to assign null values in order to overwrite initialized variables. Consider reassigning or releasing these variables instead.

Source Code Examples

Improper Validation of Array Index

Weakness ID: 129 *(Weakness Base)* **Status:** Draft

Description

Description Summary

The product uses untrusted input when calculating or using an array index, but the product does not validate or incorrectly validates the index to ensure the index references a valid position within the array.

Alternate Terms

out-of-bounds array index

index-out-of-range

array index underflow

Time of Introduction

- Implementation

Applicable Platforms

Languages

C: *(Often)*

C++: *(Often)*

Language-independent

Common Consequences

Scope	Effect
Integrity Availability	Unchecked array indexing will very likely result in the corruption of relevant memory and perhaps instructions, leading to a crash, if the values are outside of the valid memory area.
Integrity	If the memory corrupted is data, rather than instructions, the system will continue to function with improper values.
Confidentiality Integrity	Unchecked array indexing can also trigger out-of-bounds read or write operations, or operations on the wrong objects; i.e., "buffer overflows" are not always the result. This may result in the exposure or modification of sensitive data.
Integrity	If the memory accessible by the attacker can be effectively controlled, it may be possible to execute arbitrary code, as with a standard buffer overflow and possibly without the use of large inputs if a precise index can be controlled.
Integrity Availability Confidentiality	A single fault could allow either an overflow (CWE-788) or underflow (CWE-786) of the array index. What happens next will depend on the type of operation being performed out of bounds, but can expose sensitive information, cause a system crash, or possibly lead to arbitrary code execution.

Likelihood of Exploit

High

Detection Methods

Automated Static Analysis

This weakness can often be detected using automated static analysis tools. Many modern tools use data flow analysis or constraint-based techniques to minimize the number of false positives.

Automated static analysis generally does not account for environmental considerations when reporting out-of-bounds memory operations. This can make it difficult for users to determine which warnings should be investigated first. For example, an analysis tool might report array index errors that originate from command line arguments in a program that is not expected to run with setuid or other special privileges.

Effectiveness: High

This is not a perfect solution, since 100% accuracy and coverage are not feasible.

Automated Dynamic Analysis

This weakness can be detected using dynamic tools and techniques that interact with the software using large test suites with many diverse inputs, such as fuzz testing (fuzzing), robustness testing, and fault injection. The software's operation may slow down, but it should not become unstable, crash, or generate incorrect results.

Black Box

Black box methods might not get the needed code coverage within limited time constraints, and a dynamic test might not produce any noticeable side effects even if it is successful.

Demonstrative Examples

Example 1

The following C/C++ example retrieves the sizes of messages for a pop3 mail server. The message sizes are retrieved from a socket that returns in a buffer the message number and the message size, the message number (num) and size (size) are extracted from the buffer and the message size is placed into an array using the message number for the array index.

(Bad Code)

Example Language: **C**

```
/* capture the sizes of all messages */
int getsizes(int sock, int count, int *sizes) {
...
char buf[BUFFER_SIZE];
int ok;
int num, size;

// read values from socket and added to sizes array
while ((ok = gen_recv(sock, buf, sizeof(buf))) == 0)
{

// continue read from socket until buf only contains '.'
if (DOTLINE(buf))
break;
else if (sscanf(buf, "%d %d", &num, &size) == 2)
sizes[num - 1] = size;
}
...
}
```

In this example the message number retrieved from the buffer could be a value that is outside the allowable range of indices for the array and could possibly be a negative number. Without proper validation of the value to be used for the array index an array overflow could occur and could potentially lead to unauthorized access to memory addresses and system crashes. The value of the array index should be validated to ensure that it is within the allowable range of indices for the array as in the following code.

(Good Code)

Example Language: **C**

```
/* capture the sizes of all messages */
int getsizes(int sock, int count, int *sizes) {
...
char buf[BUFFER_SIZE];
int ok;
int num, size;

// read values from socket and added to sizes array
while ((ok = gen_recv(sock, buf, sizeof(buf))) == 0)
{

// continue read from socket until buf only contains '.'
if (DOTLINE(buf))
```

```
break;
else if (sscanf(buf, "%d %d", &num, &size) == 2) {
if (num > 0 && num <= (unsigned)count)
sizes[num - 1] = size;
else
/* warn about possible attempt to induce buffer overflow */
report(stderr, "Warning: ignoring bogus data for message sizes returned by server.\n");
}
}
...
}
```

Example 2

In the code snippet below, an unchecked integer value is used to reference an object in an array.

(Bad Code)

Example Language: **Java**

```
public String getValue(int index) {
return array[index];
}
```

If index is outside of the range of the array, this may result in an ArrayIndexOutOfBounds Exception being raised.

Example 3

In the following Java example the method displayProductSummary is called from a Web service servlet to retrieve product summary information for display to the user. The servlet obtains the integer value of the product number from the user and passes it to the displayProductSummary method. The displayProductSummary method passes the integer value of the product number to the getProductSummary method which obtains the product summary from the array object containing the project summaries using the integer value of the product number as the array index.

(Bad Code)

Example Language: **Java**

```
// Method called from servlet to obtain product information
public String displayProductSummary(int index) {

String productSummary = new String("");

try {
String productSummary = getProductSummary(index);

} catch (Exception ex) {...}

return productSummary;
}

public String getProductSummary(int index) {
return products[index];
}
```

In this example the integer value used as the array index that is provided by the user may be outside the allowable range of indices for the array which may provide unexpected results or may comes the application to fail. The integer value used for the array index should be validated to ensure that it is within the allowable range of indices for the array as in the following code.

(Good Code)

Example Language: **Java**

```
// Method called from servlet to obtain product information
public String displayProductSummary(int index) {

String productSummary = new String("");
```

```
try {
String productSummary = getProductSummary(index);

} catch (Exception ex) {...}

return productSummary;
}

public String getProductSummary(int index) {
String productSummary = "";

if ((index >= 0) && (index < MAX_PRODUCTS)) {
productSummary = products[index];
}
else {
System.err.println("index is out of bounds");
throw new IndexOutOfBoundsException();
}

return productSummary;
}
```

An alternative in Java would be to use one of the collection objects such as ArrayList that will automatically generate an exception if an attempt is made to access an array index that is out of bounds.

(Good Code)

Example Language: **Java**

```
ArrayList productArray = new ArrayList(MAX_PRODUCTS);
...
try {
productSummary = (String) productArray.get(index);
} catch (IndexOutOfBoundsException ex) {...}
```

Observed Examples

Reference	Description
CVE-2005-0369	large ID in packet used as array index
CVE-2001-1009	negative array index as argument to POP LIST command
CVE-2003-0721	Integer signedness error leads to negative array index
CVE-2004-1189	product does not properly track a count and a maximum number, which can lead to resultant array index overflow.
CVE-2007-5756	chain: device driver for packet-capturing software allows access to an unintended IOCTL with resultant array index error.

Potential Mitigations

Phase: Architecture and Design

Strategies: Input Validation; Libraries or Frameworks

Use an input validation framework such as Struts or the OWASP ESAPI Validation API. If you use Struts, be mindful of weaknesses covered by the CWE-101 category.

Phase: Architecture and Design

For any security checks that are performed on the client side, ensure that these checks are duplicated on the server side, in order to avoid CWE-602. Attackers can bypass the client-side checks by modifying values after the checks have been performed, or by changing the client to remove the client-side checks entirely. Then, these modified values would be submitted to the server.

Even though client-side checks provide minimal benefits with respect to server-side security, they are still useful. First, they can support intrusion detection. If the server receives input that should have been rejected by the client, then it may be an indication of an attack. Second, client-side error-checking can provide helpful feedback to the user about the expectations for valid input. Third, there may be a reduction in server-side processing time for accidental input errors, although this is typically a small savings.

Phase: Requirements

Strategy: Language Selection

Use a language with features that can automatically mitigate or eliminate out-of-bounds indexing errors.

For example, Ada allows the programmer to constrain the values of a variable and languages such as Java and Ruby will allow the programmer to handle exceptions when an out-of-bounds index is accessed.

Phase: Implementation

Strategy: Input Validation

Assume all input is malicious. Use an "accept known good" input validation strategy (i.e., use a whitelist). Reject any input that does not strictly conform to specifications, or transform it into something that does. Use a blacklist to reject any unexpected inputs and detect potential attacks.

When accessing a user-controlled array index, use a stringent range of values that are within the target array. Make sure that you do not allow negative values to be used. That is, verify the minimum as well as the maximum of the range of acceptable values.

Phase: Implementation

Be especially careful to validate your input when you invoke code that crosses language boundaries, such as from an interpreted language to native code. This could create an unexpected interaction between the language boundaries. Ensure that you are not violating any of the expectations of the language with which you are interfacing. For example, even though Java may not be susceptible to buffer overflows, providing a large argument in a call to native code might trigger an overflow.

Weakness Ordinalities

Ordinality	Description
Resultant	The most common condition situation leading to unchecked array indexing is the use of loop index variables as buffer indexes. If the end condition for the loop is subject to a flaw, the index can grow or shrink unbounded, therefore causing a buffer overflow or underflow. Another common situation leading to this condition is the use of a function's return value, or the resulting value of a calculation directly as an index in to a buffer.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	20	Improper Input Validation	**Development Concepts (primary)699 Research Concepts (primary)1000**
ChildOf	Category	189	Numeric Errors	Development Concepts699
ChildOf	Category	633	Weaknesses that Affect Memory	**Resource-specific Weaknesses (primary)631**
ChildOf	Category	738	CERT C Secure Coding Section 04 - Integers (INT)	**Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734**
ChildOf	Category	740	CERT C Secure Coding Section 06 - Arrays (ARR)	Weaknesses Addressed by the CERT C Secure Coding Standard734
ChildOf	Category	802	2010 Top 25 - Risky Resource Management	**Weaknesses in the 2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800**
CanPrecede	Weakness Class	119	Failure to Constrain Operations within the Bounds of a Memory Buffer	Research Concepts1000
CanPrecede	Weakness Variant	789	Uncontrolled Memory Allocation	Research Concepts1000
PeerOf	Weakness Base	124	Buffer Underwrite ('Buffer Underflow')	Research Concepts1000

Theoretical Notes

An improperly validated array index might lead directly to the always-incorrect behavior of "access of array using out-of-bounds index."

Affected Resources

- Memory

f Causal Nature

Explicit

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
CLASP			Unchecked array indexing
PLOVER			INDEX - Array index overflow
CERT C Secure Coding	ARR00-C		Understand how arrays work
CERT C Secure Coding	ARR30-C		Guarantee that array indices are within the valid range
CERT C Secure Coding	ARR38-C		Do not add or subtract an integer to a pointer if the resulting value does not refer to a valid array element
CERT C Secure Coding	INT32-C		Ensure that operations on signed integers do not result in overflow

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
100	Overflow Buffers	

References

[REF-11] M. Howard and D. LeBlanc. "Writing Secure Code". Chapter 5, "Array Indexing Errors" Page 144. 2nd Edition. Microsoft. 2002.

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	CLASP		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Sean Eidemiller	Cigital	External
	added/updated demonstrative examples		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Alternate Terms, Applicable Platforms, Common Consequences, Relationships, Other Notes, Taxonomy Mappings, Weakness Ordinalities		
2008-11-24	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2009-01-12	CWE Content Team	MITRE	Internal
	updated Common Consequences		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Description, Name, Relationships		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Observed Examples, Other Notes, Potential Mitigations, Theoretical Notes, Weakness Ordinalities		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Demonstrative Examples, Detection Factors, Likelihood of Exploit, Potential Mitigations, References, Related Attack Patterns, Relationships		
2010-04-05	CWE Content Team	MITRE	Internal
	updated Related Attack Patterns		
Previous Entry Names			
Change Date	**Previous Entry Name**		
2009-10-29	Unchecked Array Indexing		

BACK TO TOP

Scanned Languages

Language	Hash Number	Change Date
CPP	4541647240435660	1/6/2025
Common	0105849645654507	1/6/2025