vul_files_4 Scan Report

Project Name	vul_files_4
Scan Start	Monday, January 6, 2025 2:20:08 PM
Preset	Checkmarx Default
Scan Time	02h:25m:55s
Lines Of Code Scanned	298323
Files Scanned	126
Report Creation Time	Monday, January 6, 2025 6:40:43 PM
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5
Team	CxServer
Checkmarx Version	8.7.0
Scan Type	Full
Source Origin	LocalPath
Density	10/1000 (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

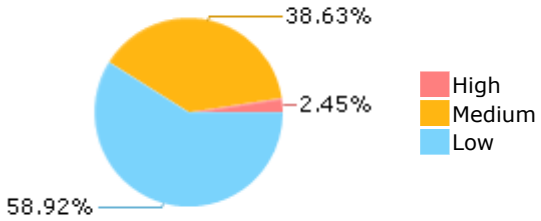
Results Limit

Results limit per query was set to 50

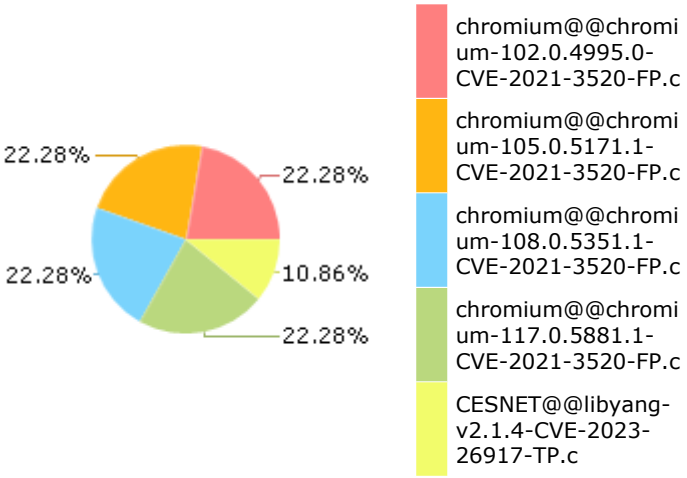
Selected Queries

Selected queries are listed in Result Summary

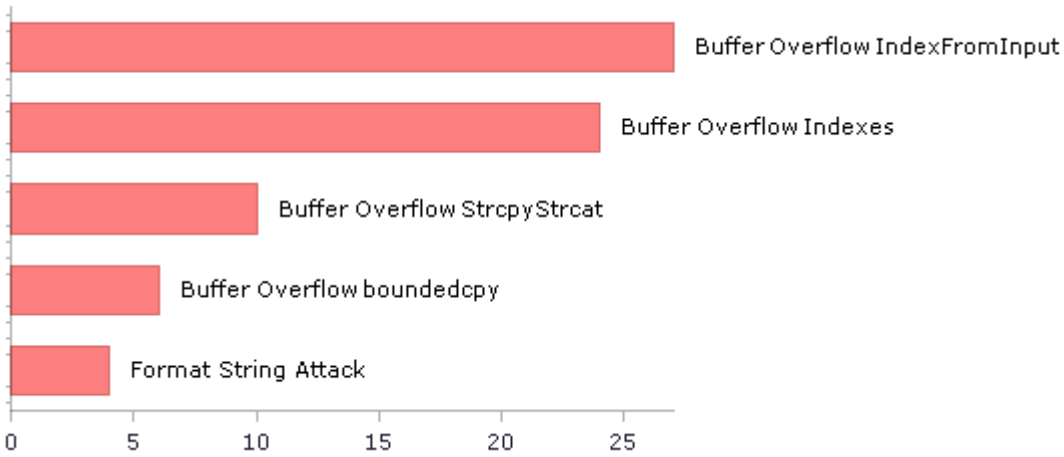
Result Summary



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	580	346
A2-Broken Authentication	App. Specific	EASY	COMMON	AVERAGE	SEVERE	App. Specific	98	98
A3-Sensitive Data Exposure	App. Specific	AVERAGE	WIDESPREAD	AVERAGE	SEVERE	App. Specific	6	3
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	11	9
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	441	441
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	5	5
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	11	9
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	441	441
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	32	32
PCI DSS (3.2) - 6.5.2 - Buffer overflows	246	216
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.	18	18
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.	15	15
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.	19	16
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.	86	86
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.	5	5

***** 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)	111	111
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)	6	3
SC-17 Public Key Infrastructure Certificates (P1)	0	0
SC-18 Mobile Code (P2)	0	0
SC-23 Session Authenticity (P1)*	6	6
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)*	777	388
SC-8 Transmission Confidentiality and Integrity (P1)	0	0
SI-10 Information Input Validation (P1)*	871	841
SI-11 Error Handling (P2)*	215	215
SI-15 Information Output Filtering (P0)	0	0
SI-16 Memory Protection (P1)	33	33

***** 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

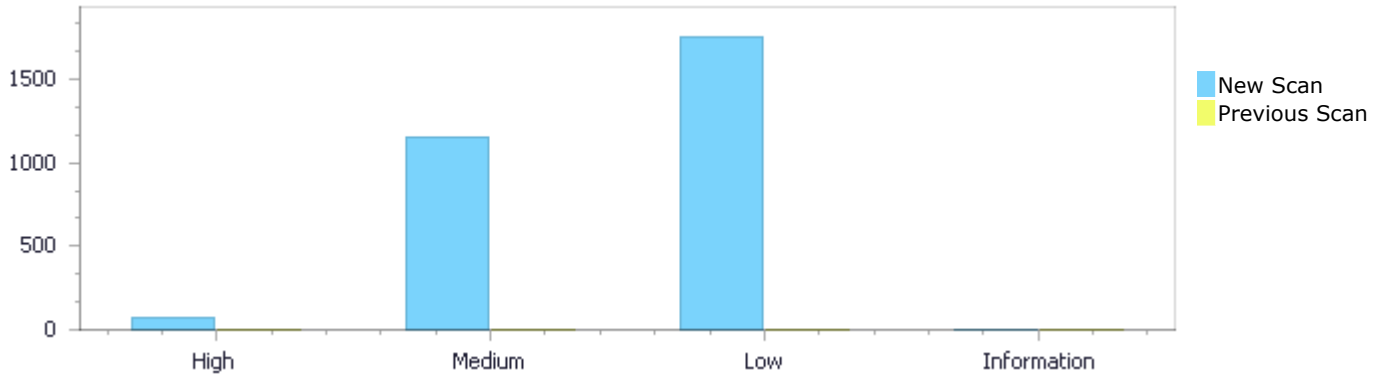
Scan Summary - Custom

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	73	1,152	1,757	0	2,982
Recurrent Issues	0	0	0	0	0
Total	73	1,152	1,757	0	2,982
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	73	1,152	1,757	0	2,982
Urgent	0	0	0	0	0
Proposed Not Exploitable	0	0	0	0	0
Total	73	1,152	1,757	0	2,982

Result Summary

Vulnerability Type	Occurrences	Severity
Buffer Overflow IndexFromInput	27	High
Buffer Overflow Indexes	24	High
Buffer Overflow StrcpyStrcat	10	High
Buffer Overflow boundedcpy	6	High
Format String Attack	4	High

String Termination Error	2	High
Dangerous Functions	441	Medium
Use of Zero Initialized Pointer	415	Medium
Buffer Overflow boundcpy WrongSizeParam	162	Medium
Memory Leak	55	Medium
Buffer Overflow AddressOfLocalVarReturned	30	Medium
MemoryFree on StackVariable	15	Medium
Stored Buffer Overflow fgets	8	Medium
Wrong Size t Allocation	8	Medium
Inadequate Encryption Strength	6	Medium
Use After Free	6	Medium
Environment Injection	5	Medium
Double Free	1	Medium
Unchecked Array Index	804	Low
NULL Pointer Dereference	256	Low
Unchecked Return Value	215	Low
Use of Sizeof On a Pointer Type	177	Low
Improper Resource Access Authorization	80	Low
Sizeof Pointer Argument	78	Low
TOCTOU	44	Low
Potential Off by One Error in Loops	32	Low
Incorrect Permission Assignment For Critical Resources	18	Low
Arithmenic Operation On Boolean	15	Low
Exposure of System Data to Unauthorized Control Sphere	13	Low
Potential Path Traversal	11	Low
Heuristic Buffer Overflow malloc	8	Low
Reliance on DNS Lookups in a Decision	6	Low

10 Most Vulnerable Files

High and Medium Vulnerabilities

File Name	Issues Found
c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp	51
chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	39
chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	39
chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	39
bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c	38
bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	37
bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	36
CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c	35
bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c	33
bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c	33

Scan Results Details

Buffer Overflow IndexFromInput

Query Path:
CPP\Cx\CPP Buffer Overflow\Buffer Overflow IndexFromInput Version:1

Categories

OWASP Top 10 2017: A1-Injection

Description

Buffer Overflow IndexFromInput\Path 1:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=31
Status	New

The size of the buffer used by match in n, at line 980 of ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1161 of ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c	ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c
Line	1161	1081
Object	argv	n

Code Snippet

File Name ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c

Method int main(int argc, char **argv)

```
....
1161.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c

Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1081.                         if (strncmpcanon(sp, out->sub[pc->n].sp,
i))
```

Buffer Overflow IndexFromInput\Path 2:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&

	pathid=32
Status	New

The size of the buffer used by match in n, at line 980 of ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1161 of ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c	ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c
Line	1161	1084
Object	argv	n

Code Snippet

File Name ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c

Method int main(int argc, char **argv)

```
....
1161.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c

Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1084.                          if (strncmp(sp, out->sub[pc->n].sp, i))
```

Buffer Overflow IndexFromInput\Path 3:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=33
Status	New

The size of the buffer used by match in n, at line 980 of ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1161 of ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c	ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c
Line	1161	1132
Object	argv	n

Code Snippet

File Name ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c

Method	int main(int argc, char **argv)

```
....
1161.  int main(int argc, char **argv)
```

File Name	ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c
Method	static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1132.                  out->sub[pc->n].sp = sp;
```

Buffer Overflow IndexFromInput\Path 4:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=34
Status	New

The size of the buffer used by match in n, at line 980 of ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1161 of ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c	ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c
Line	1161	1136
Object	argv	n

Code Snippet	
File Name	ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c
Method	int main(int argc, char **argv)

```
....
1161.  int main(int argc, char **argv)
```

File Name	ccxvii@@mujs-1.0.7-CVE-2022-30974-TP.c
Method	static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1136.                  out->sub[pc->n].ep = sp;
```

Buffer Overflow IndexFromInput\Path 5:

Severity	High
Result State	To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=35
Status	New

The size of the buffer used by match in n, at line 988 of ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1164 of ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c	ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c
Line	1164	1084
Object	argv	n

Code Snippet

File Name ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c

Method int main(int argc, char **argv)

```
....
1164.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c

Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1084.                         if (strncmpcanon(sp, out->sub[pc->n].sp,
i))
```

Buffer Overflow IndexFromInput\Path 6:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=36
Status	New

The size of the buffer used by match in n, at line 988 of ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1164 of ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c	ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c
Line	1164	1087
Object	argv	n

Code Snippet
File Name ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c
Method int main(int argc, char **argv)

```
....
1164.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c

Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1087.                          if (strncmp(sp, out->sub[pc->n].sp, i))
```

Buffer Overflow IndexFromInput\Path 7:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=37
Status	New

The size of the buffer used by match in n, at line 988 of ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1164 of ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c	ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c
Line	1164	1135
Object	argv	n

Code Snippet
File Name ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c
Method int main(int argc, char **argv)

```
....
1164.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c

Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1135.                  out->sub[pc->n].sp = sp;
```

Buffer Overflow IndexFromInput\Path 8:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=38
Status	New

The size of the buffer used by match in n, at line 988 of ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1164 of ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c	ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c
Line	1164	1139
Object	argv	n

Code Snippet

File Name ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c

Method int main(int argc, char **argv)

```
....
1164.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.0.8-CVE-2022-30974-TP.c

Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1139.                   out->sub[pc->n].ep = sp;
```

Buffer Overflow IndexFromInput\Path 9:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=39
Status	New

The size of the buffer used by match in n, at line 994 of ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1170 of ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c	ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c
Line	1170	1090

Object	argv	n

Code Snippet
File Name ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c
Method int main(int argc, char **argv)

```
....
1170.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c

Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1090.                          if (strncmpcanon(sp, out->sub[pc->n].sp,
i))
```

Buffer Overflow IndexFromInput\Path 10:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=40
Status	New

The size of the buffer used by match in n, at line 994 of ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1170 of ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c	ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c
Line	1170	1093
Object	argv	n

Code Snippet
File Name ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c
Method int main(int argc, char **argv)

```
....
1170.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c

Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1093.                           if (strncmp(sp, out->sub[pc->n].sp, i))
```

Buffer Overflow IndexFromInput\Path 11:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=41
Status	New

The size of the buffer used by match in n, at line 994 of ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1170 of ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c	ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c
Line	1170	1141
Object	argv	n

Code Snippet

File Name ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c

Method int main(int argc, char **argv)

```
....
1170.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c

Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1141.                   out->sub[pc->n].sp = sp;
```

Buffer Overflow IndexFromInput\Path 12:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=42
Status	New

The size of the buffer used by match in n, at line 994 of ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1170 of ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c	ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c
Line	1170	1145
Object	argv	n

Code Snippet
File Name ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c
Method int main(int argc, char **argv)

```
....
1170.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.1.0-CVE-2022-30974-TP.c

Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1145.                  out->sub[pc->n].ep = sp;
```

Buffer Overflow IndexFromInput\Path 13:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=43
Status	New

The size of the buffer used by match in n, at line 1009 of ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1185 of ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c	ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c
Line	1185	1105
Object	argv	n

Code Snippet
File Name ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c
Method int main(int argc, char **argv)

```
....
1185.  int main(int argc, char **argv)
```

File Name	ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c
Method	static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1105.                          if (strncmpcanon(sp, out->sub[pc->n].sp,
i))
```

Buffer Overflow IndexFromInput\Path 14:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=44
Status	New

The size of the buffer used by match in n, at line 1009 of ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1185 of ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c	ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c
Line	1185	1108
Object	argv	n

Code Snippet

File Name	ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c
Method	int main(int argc, char **argv)

```
....
1185.  int main(int argc, char **argv)
```

File Name	ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c
Method	static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1108.                          if (strncmp(sp, out->sub[pc->n].sp, i))
```

Buffer Overflow IndexFromInput\Path 15:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=45
Status	New

The size of the buffer used by match in n, at line 1009 of ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1185 of ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c	ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c
Line	1185	1156
Object	argv	n

Code Snippet
File Name ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c
Method int main(int argc, char **argv)

```
....
1185.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c
Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1156.                   out->sub[pc->n].sp = sp;
```

Buffer Overflow IndexFromInput\Path 16:

Severity High
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=46
Status New

The size of the buffer used by match in n, at line 1009 of ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1185 of ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c	ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c
Line	1185	1160
Object	argv	n

Code Snippet
File Name ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c
Method int main(int argc, char **argv)

```
....
1185.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.1.3-CVE-2022-30974-TP.c

Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1160.                  out->sub[pc->n].ep = sp;
```

Buffer Overflow IndexFromInput\Path 17:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=47
Status	New

The size of the buffer used by match in n, at line 1013 of ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1189 of ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c	ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c
Line	1189	1109
Object	argv	n

Code Snippet

File Name ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c

Method int main(int argc, char **argv)

```
....
1189.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c

Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1109.                      if (strncmpcanon(sp, out->sub[pc->n].sp,
i))
```

Buffer Overflow IndexFromInput\Path 18:

Severity	High
Result State	To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=48
Status	New

The size of the buffer used by match in n, at line 1013 of ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1189 of ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c	ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c
Line	1189	1112
Object	argv	n

Code Snippet

File Name ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c

Method int main(int argc, char **argv)

```
....
1189.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c

Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1112.                         if (strncmp(sp, out->sub[pc->n].sp, i))
```

Buffer Overflow IndexFromInput\Path 19:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=49
Status	New

The size of the buffer used by match in n, at line 1013 of ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1189 of ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c	ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c
Line	1189	1160
Object	argv	n

Code Snippet
File Name ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c
Method int main(int argc, char **argv)

```
....
1189.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c

Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1160.                  out->sub[pc->n].sp = sp;
```

Buffer Overflow IndexFromInput\Path 20:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=50
Status	New

The size of the buffer used by match in n, at line 1013 of ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1189 of ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c	ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c
Line	1189	1164
Object	argv	n

Code Snippet
File Name ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c
Method int main(int argc, char **argv)

```
....
1189.  int main(int argc, char **argv)
```

File Name ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c

Method static int match(Reinst *pc, const char *sp, const char *bol, int flags, Resub *out, int depth)

```
....
1164.                  out->sub[pc->n].ep = sp;
```

Buffer Overflow IndexFromInput\Path 21:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=51
Status	New

The size of the buffer used by Instance_DidCreate in i, at line 86 of chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that Instance_DidCreate passes to getenv, at line 86 of chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Line	127	147
Object	getenv	i

Code Snippet
File Name chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
127.      const char* next_arg = getenv(arg_name);
....
147.      PSInstanceTrace("argv[%d] '%s'\n", i, si->argv_[i]);
```

Buffer Overflow IndexFromInput\Path 22:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=52
Status	New

The size of the buffer used by Instance_DidCreate in i, at line 86 of chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that Instance_DidCreate passes to getenv, at line 86 of chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Line	127	147
Object	getenv	i

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
127.      const char* next_arg = getenv(arg_name);
....
147.      PSInstanceTrace("argv[%d] '%s'\n", i, si->argv_[i]);
```

Buffer Overflow IndexFromInput\Path 23:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=53
Status	New

The size of the buffer used by Instance_DidCreate in i, at line 86 of chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that Instance_DidCreate passes to getenv, at line 86 of chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Line	127	147
Object	getenv	i

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
127.      const char* next_arg = getenv(arg_name);
....
147.      PSInstanceTrace("argv[%d] '%s'\n", i, si->argv_[i]);
```

Buffer Overflow IndexFromInput\Path 24:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=54
Status	New

The size of the buffer used by Instance_DidCreate in i, at line 86 of chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that Instance_DidCreate passes to getenv, at line 86 of chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Line	127	147
Object	getenv	i

Code Snippet	
File Name	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Method	static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
127.      const char* next_arg = getenv(arg_name);
....
147.      PSInstanceTrace("argv[%d] '%s'\n", i, si->argv_[i]);
```

Buffer Overflow IndexFromInput\Path 25:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=55
Status	New

The size of the buffer used by Instance_DidCreate in i, at line 86 of chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that Instance_DidCreate passes to getenv, at line 86 of chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Line	127	147
Object	getenv	i

Code Snippet	
File Name	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Method	static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
127.      const char* next_arg = getenv(arg_name);
....
147.      PSInstanceTrace("argv[%d] '%s'\n", i, si->argv_[i]);
```

Buffer Overflow IndexFromInput\Path 26:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=56
Status	New

The size of the buffer used by get_exe_path in size, at line 50 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that get_exe_path passes to path_buf, at line 50 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination

File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	53	59
Object	path_buf	size

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method get_exe_path(char *path_buf, unsigned path_buf_size)

```
....
53.      ssize_t size = readlink("/proc/self/exe", path_buf,
path_buf_size - 1);
....
59.      path_buf[size] = '\0';
```

Buffer Overflow IndexFromInput\Path 27:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=57
Status	New

The size of the buffer used by get_exe_path in size, at line 50 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that get_exe_path passes to path_buf, at line 50 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	53	59
Object	path_buf	size

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method get_exe_path(char *path_buf, unsigned path_buf_size)

```
....
53.      ssize_t size = readlink("/proc/self/exe", path_buf,
path_buf_size - 1);
....
59.      path_buf[size] = '\0';
```

Buffer Overflow Indexes

Query Path:
CPP\Cx\CPP Buffer Overflow\Buffer Overflow Indexes Version:1

Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.2 - Buffer overflows
NIST SP 800-53: SI-10 Information Input Validation (P1)
OWASP Top 10 2017: A1-Injection

Description

Buffer Overflow Indexes\Path 1:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1
Status	New

The size of the buffer used by xmlNanoFTPGetSocket in buf, at line 1818 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 2055 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	2055	1850
Object	argv	buf

Code Snippet

File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c

Method int main(int argc, char **argv) {

```
....
2055.  int main(int argc, char **argv) {
```

File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c

Method xmlNanoFTPGetSocket(void *ctx, const char *filename) {

```
....
1850.      buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 2:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2
Status	New

The size of the buffer used by xmlNanoFTPGetSocket in sizeof, at line 1818 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 2055 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination

File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	2055	1850
Object	argv	sizeof

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method int main(int argc, char **argv) {

```
....
2055.  int main(int argc, char **argv) {
```

File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPGetSocket(void *ctx, const char *filename) {

```
....
1850.      buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 3:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=3
Status	New

The size of the buffer used by xmlNanoFTPList in buf, at line 1708 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 2055 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	2055	1735
Object	argv	buf

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method int main(int argc, char **argv) {

```
....
2055.  int main(int argc, char **argv) {
```

File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1735.       buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 4:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=4
Status	New

The size of the buffer used by xmlNanoFTPList in sizeof, at line 1708 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 2055 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	2055	1735
Object	argv	sizeof

Code Snippet

File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c

Method int main(int argc, char **argv) {

```
....
2055.  int main(int argc, char **argv) {
```

File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c

Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1735.       buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 5:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=5
Status	New

The size of the buffer used by xmlNanoFTPGetSocket in buf, at line 1818 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 2055 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination

File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	2055	1850
Object	argv	buf

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method int main(int argc, char **argv) {

```
....
2055.  int main(int argc, char **argv) {
```

File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPGetSocket(void *ctx, const char *filename) {

```
....
1850.      buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 6:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=6
Status	New

The size of the buffer used by xmlNanoFTPGetSocket in sizeof, at line 1818 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 2055 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	2055	1850
Object	argv	sizeof

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method int main(int argc, char **argv) {

```
....
2055.  int main(int argc, char **argv) {
```

File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPGetSocket(void *ctx, const char *filename) {

```
....
1850.       buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 7:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=7
Status	New

The size of the buffer used by xmlNanoFTPList in buf, at line 1708 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 2055 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	2055	1735
Object	argv	buf

Code Snippet

File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c

Method int main(int argc, char **argv) {

```
....
2055.  int main(int argc, char **argv) {
```

File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c

Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1735.       buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 8:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=8
Status	New

The size of the buffer used by xmlNanoFTPList in sizeof, at line 1708 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 2055 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination

File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	2055	1735
Object	argv	sizeof

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method int main(int argc, char **argv) {

```
....
2055.  int main(int argc, char **argv) {
```

File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1735.      buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 9:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=9
Status	New

The size of the buffer used by xmlNanoFTPGetSocket in buf, at line 1714 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1939 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	1939	1743
Object	argv	buf

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method int main(int argc, char **argv) {

```
....
1939.  int main(int argc, char **argv) {
```

File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPGetSocket(void *ctx, const char *filename) {

```
....
1743.       buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 10:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=10
Status	New

The size of the buffer used by xmlNanoFTPGetSocket in sizeof, at line 1714 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1939 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	1939	1743
Object	argv	sizeof

Code Snippet

File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c

Method int main(int argc, char **argv) {

```
....
1939.  int main(int argc, char **argv) {
```

File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c

Method xmlNanoFTPGetSocket(void *ctx, const char *filename) {

```
....
1743.       buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 11:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=11
Status	New

The size of the buffer used by xmlNanoFTPList in buf, at line 1613 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1939 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination

File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	1939	1640
Object	argv	buf

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method int main(int argc, char **argv) {

```
....
1939.  int main(int argc, char **argv) {
```

File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1640.      buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 12:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=12
Status	New

The size of the buffer used by xmlNanoFTPList in sizeof, at line 1613 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 1939 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	1939	1640
Object	argv	sizeof

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method int main(int argc, char **argv) {

```
....
1939.  int main(int argc, char **argv) {
```

File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1640.       buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 13:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=13
Status	New

The size of the buffer used by xmlNanoFTPConnect in buf, at line 832 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPInit passes to getenv, at line 163 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	190	1005
Object	getenv	buf

Code Snippet

File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c

Method xmlNanoFTPInit(void) {

```
....
190.       env = getenv("ftp_proxy_user");
```

File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c

Method xmlNanoFTPConnect(void *ctx) {

```
....
1005.              buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 14:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=14
Status	New

The size of the buffer used by xmlNanoFTPConnect in sizeof, at line 832 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPInit passes to getenv, at line 163 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination

File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	190	1005
Object	getenv	sizeof

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPInit(void) {

```
....
190.      env = getenv("ftp_proxy_user");
```

File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
1005.             buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 15:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=15
Status	New

The size of the buffer used by xmlNanoFTPConnect in buf, at line 832 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPInit passes to getenv, at line 163 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	194	1028
Object	getenv	buf

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPInit(void) {

```
....
194.      env = getenv("ftp_proxy_password");
```

File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
1028.                    buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 16:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=16
Status	New

The size of the buffer used by xmlNanoFTPConnect in sizeof, at line 832 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPInit passes to getenv, at line 163 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	194	1028
Object	getenv	sizeof

Code Snippet

File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c

Method xmlNanoFTPInit(void) {

```
....
194.      env = getenv("ftp_proxy_password");
```

File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c

Method xmlNanoFTPConnect(void *ctx) {

```
....
1028.                    buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 17:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=17
Status	New

The size of the buffer used by xmlNanoFTPConnect in buf, at line 832 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPInit passes to getenv, at line 163 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination

File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	190	1005
Object	getenv	buf

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPInit(void) {

```
....
190.      env = getenv("ftp_proxy_user");
```

File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
1005.             buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 18:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=18
Status	New

The size of the buffer used by xmlNanoFTPConnect in sizeof, at line 832 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPInit passes to getenv, at line 163 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	190	1005
Object	getenv	sizeof

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPInit(void) {

```
....
190.      env = getenv("ftp_proxy_user");
```

File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
1005.               buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 19:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=19
Status	New

The size of the buffer used by xmlNanoFTPConnect in buf, at line 832 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPInit passes to getenv, at line 163 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	194	1028
Object	getenv	buf

Code Snippet

File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c

Method xmlNanoFTPInit(void) {

```
....
194.      env = getenv("ftp_proxy_password");
```

File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c

Method xmlNanoFTPConnect(void *ctx) {

```
....
1028.                     buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 20:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=20
Status	New

The size of the buffer used by xmlNanoFTPConnect in sizeof, at line 832 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPInit passes to getenv, at line 163 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination

File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	194	1028
Object	getenv	sizeof

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPInit(void) {

```
....
194.      env = getenv("ftp_proxy_password");
```

File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
1028.                     buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 21:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=21
Status	New

The size of the buffer used by xmlNanoFTPConnect in buf, at line 771 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPInit passes to getenv, at line 154 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	181	944
Object	getenv	buf

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPInit(void) {

```
....
181.      env = getenv("ftp_proxy_user");
```

File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
944.              buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 22:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=22
Status	New

The size of the buffer used by xmlNanoFTPConnect in sizeof, at line 771 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPInit passes to getenv, at line 154 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	181	944
Object	getenv	sizeof

Code Snippet

File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c

Method xmlNanoFTPInit(void) {

```
....
181.      env = getenv("ftp_proxy_user");
```

File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c

Method xmlNanoFTPConnect(void *ctx) {

```
....
944.              buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 23:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=23
Status	New

The size of the buffer used by xmlNanoFTPConnect in buf, at line 771 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPInit passes to getenv, at line 154 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination

File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	185	964
Object	getenv	buf

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPInit(void) {

```
....
185.      env = getenv("ftp_proxy_password");
```

File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
964.                      buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow Indexes\Path 24:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=24
Status	New

The size of the buffer used by xmlNanoFTPConnect in sizeof, at line 771 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPInit passes to getenv, at line 154 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	185	964
Object	getenv	sizeof

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPInit(void) {

```
....
185.      env = getenv("ftp_proxy_password");
```

File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
964.                        buf[sizeof(buf) - 1] = 0;
```

Buffer Overflow StrcpyStrcat

Query Path:
CPP\Cx\CPP Buffer Overflow\Buffer Overflow StrcpyStrcat Version:1

Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.2 - Buffer overflows
NIST SP 800-53: SI-10 Information Input Validation (P1)
OWASP Top 10 2017: A1-Injection

Description

Buffer Overflow StrcpyStrcat\Path 1:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=62
Status	New

The size of the buffer used by lysp_stmt_type_pattern_modifier in pat, at line 780 of CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that lysp_stmt_type_pattern_modifier passes to pat, at line 780 of CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c, to overwrite the target buffer.

	Source	Destination
File	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Line	780	800
Object	pat	pat

Code Snippet
File Name CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Method lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, const char **pat, struct lysp_ext_instance **exts)

```
....
780.  lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const
struct lysp_stmt *stmt, const char **pat, struct lysp_ext_instance
**exts)
....
800.      strcpy(buf, *pat);
```

Buffer Overflow StrcpyStrcat\Path 2:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=63
Status	New

The size of the buffer used by lysp_stmt_type_pattern_modifier in Pointer, at line 780 of CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that lysp_stmt_type_pattern_modifier passes to pat, at line 780 of CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c, to overwrite the target buffer.

	Source	Destination
File	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Line	780	800
Object	pat	Pointer

Code Snippet
File Name CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Method lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, const char **pat, struct lysp_ext_instance **exts)

```
....
780.  lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const
struct lysp_stmt *stmt, const char **pat, struct lysp_ext_instance
**exts)
....
800.      strcpy(buf, *pat);
```

Buffer Overflow StrcpyStrcat\Path 3:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=64
Status	New

The size of the buffer used by lysp_stmt_type_pattern_modifier in pat, at line 780 of CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that lysp_stmt_type_pattern_modifier passes to pat, at line 780 of CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c, to overwrite the target buffer.

	Source	Destination
File	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Line	780	800
Object	pat	pat

Code Snippet
File Name CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Method lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, const char **pat, struct lysp_ext_instance **exts)

```
....
780.  lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const
struct lysp_stmt *stmt, const char **pat, struct lysp_ext_instance
**exts)
....
800.      strcpy(buf, *pat);
```

Buffer Overflow StrcpyStrcat\Path 4:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=65
Status	New

The size of the buffer used by lysp_stmt_type_pattern_modifier in Pointer, at line 780 of CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that lysp_stmt_type_pattern_modifier passes to pat, at line 780 of CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c, to overwrite the target buffer.

	Source	Destination
File	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Line	780	800
Object	pat	Pointer

Code Snippet
File Name CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Method lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, const char **pat, struct lysp_ext_instance **exts)

```
....
780.  lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const
struct lysp_stmt *stmt, const char **pat, struct lysp_ext_instance
**exts)
....
800.      strcpy(buf, *pat);
```

Buffer Overflow StrcpyStrcat\Path 5:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=66
Status	New

The size of the buffer used by lysp_stmt_type_pattern_modifier in pat, at line 1035 of CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that lysp_stmt_type_pattern_modifier passes to pat, at line 1035 of CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c, to overwrite the target buffer.

	Source	Destination

File	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Line	1035	1056
Object	pat	pat

Code Snippet
File Name CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Method lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, const char **pat,

```
....
1035.  lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const
struct lysp_stmt *stmt, const char **pat,
....
1056.      strcpy(buf, *pat);
```

Buffer Overflow StrcpyStrcat\Path 6:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=67
Status	New

The size of the buffer used by lysp_stmt_type_pattern_modifier in Pointer, at line 1035 of CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that lysp_stmt_type_pattern_modifier passes to pat, at line 1035 of CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c, to overwrite the target buffer.

	Source	Destination
File	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Line	1035	1056
Object	pat	Pointer

Code Snippet
File Name CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Method lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, const char **pat,

```
....
1035.  lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const
struct lysp_stmt *stmt, const char **pat,
....
1056.      strcpy(buf, *pat);
```

Buffer Overflow StrcpyStrcat\Path 7:

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

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=68
Status	New

The size of the buffer used by lysp_stmt_type_pattern_modifier in pat, at line 780 of CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that lysp_stmt_type_pattern_modifier passes to pat, at line 780 of CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c, to overwrite the target buffer.

	Source	Destination
File	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Line	780	800
Object	pat	pat

Code Snippet

File Name CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c

Method lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, const char **pat, struct lysp_ext_instance **exts)

```
....
780.  lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const
struct lysp_stmt *stmt, const char **pat, struct lysp_ext_instance
**exts)
....
800.      strcpy(buf, *pat);
```

Buffer Overflow StrcpyStrcat\Path 8:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=69
Status	New

The size of the buffer used by lysp_stmt_type_pattern_modifier in Pointer, at line 780 of CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that lysp_stmt_type_pattern_modifier passes to pat, at line 780 of CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c, to overwrite the target buffer.

	Source	Destination
File	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Line	780	800
Object	pat	Pointer

Code Snippet

File Name CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c

Method lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, const char **pat, struct lysp_ext_instance **exts)

```
....
780.  lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const
struct lysp_stmt *stmt, const char **pat, struct lysp_ext_instance
**exts)
....
800.      strcpy(buf, *pat);
```

Buffer Overflow StrcpyStrcat\Path 9:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=70
Status	New

The size of the buffer used by lysp_stmt_type_pattern_modifier in pat, at line 1093 of CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that lysp_stmt_type_pattern_modifier passes to pat, at line 1093 of CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c, to overwrite the target buffer.

	Source	Destination
File	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Line	1093	1114
Object	pat	pat

Code Snippet
File Name CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Method lysp_stmt_type_pattern_modifier(struct lysp_ctx *ctx, const struct lysp_stmt *stmt, const char **pat,

```
....
1093.  lysp_stmt_type_pattern_modifier(struct lysp_ctx *ctx, const
struct lysp_stmt *stmt, const char **pat,
....
1114.      strcpy(buf, *pat);
```

Buffer Overflow StrcpyStrcat\Path 10:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=71
Status	New

The size of the buffer used by lysp_stmt_type_pattern_modifier in Pointer, at line 1093 of CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that lysp_stmt_type_pattern_modifier passes to pat, at line 1093 of CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c, to overwrite the target buffer.

	Source	Destination
File	CESNET@@libyang-v2.1.4-CVE-2023-	CESNET@@libyang-v2.1.4-CVE-2023-

	26917-TP.c	26917-TP.c
Line	1093	1114
Object	pat	Pointer

Code Snippet
File Name CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Method lysp_stmt_type_pattern_modifier(struct lysp_ctx *ctx, const struct lysp_stmt *stmt, const char **pat,

```
....
1093.  lysp_stmt_type_pattern_modifier(struct lysp_ctx *ctx, const
struct lysp_stmt *stmt, const char **pat,
....
1114.      strcpy(buf, *pat);
```

Buffer Overflow boundedcpy

Query Path:
CPP\Cx\CPP Buffer Overflow\Buffer Overflow boundedcpy Version:1

Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.2 - Buffer overflows
NIST SP 800-53: SI-10 Information Input Validation (P1)
OWASP Top 10 2017: A1-Injection

Description
Buffer Overflow boundedcpy\Path 1:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=25
Status	New

The size parameter h_length in line 832 in file chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c is influenced by the user input getenv in line 163 in file chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c. This may lead to a buffer overflow vulnerability, which may in turn result in malicious code execution.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	181	928
Object	getenv	h_length

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPInit(void) {

```
....
181.      env = getenv("ftp_proxy");
```

File Name	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method	xmlNanoFTPConnect(void *ctx) {

```
....
928.              hp->h_addr_list[0], hp->h_length);
```

Buffer Overflow boundedcpy\Path 2:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=26
Status	New

The size parameter h_length in line 832 in file chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c is influenced by the user input getenv in line 163 in file chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c. This may lead to a buffer overflow vulnerability, which may in turn result in malicious code execution.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	185	928
Object	getenv	h_length

Code Snippet

File Name	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method	xmlNanoFTPInit(void) {

```
....
185.        env = getenv("FTP_PROXY");
```

File Name	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method	xmlNanoFTPConnect(void *ctx) {

```
....
928.              hp->h_addr_list[0], hp->h_length);
```

Buffer Overflow boundedcpy\Path 3:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=27
Status	New

The size parameter h_length in line 832 in file chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c is influenced by the user input getenv in line 163 in file chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c. This may lead to a buffer overflow vulnerability, which may in turn result in malicious code execution.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	181	928
Object	getenv	h_length

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPInit(void) {

```
....
181.        env = getenv("ftp_proxy");
```

File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
928.                hp->h_addr_list[0], hp->h_length);
```

Buffer Overflow boundedcpy\Path 4:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=28
Status	New

The size parameter h_length in line 832 in file chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c is influenced by the user input getenv in line 163 in file chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c. This may lead to a buffer overflow vulnerability, which may in turn result in malicious code execution.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	185	928
Object	getenv	h_length

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPInit(void) {

```
....
185.        env = getenv("FTP_PROXY");
```

File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c

Method xmlNanoFTPConnect(void *ctx) {

```
....
928.              hp->h_addr_list[0], hp->h_length);
```

Buffer Overflow boundedcpy\Path 5:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=29
Status	New

The size parameter h_length in line 771 in file chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c is influenced by the user input getenv in line 154 in file chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c. This may lead to a buffer overflow vulnerability, which may in turn result in malicious code execution.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	172	867
Object	getenv	h_length

Code Snippet

File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c

Method xmlNanoFTPInit(void) {

```
....
172.      env = getenv("ftp_proxy");
```

File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c

Method xmlNanoFTPConnect(void *ctx) {

```
....
867.              hp->h_addr_list[0], hp->h_length);
```

Buffer Overflow boundedcpy\Path 6:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=30
Status	New

The size parameter h_length in line 771 in file chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c is influenced by the user input getenv in line 154 in file chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c. This may lead to a buffer overflow vulnerability, which may in turn result in malicious code execution.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	176	867
Object	getenv	h_length

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPInit(void) {

```
....
176.        env = getenv("FTP_PROXY");
```

File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
867.                hp->h_addr_list[0], hp->h_length);
```

Format String Attack

Query Path:
CPP\Cx\CPP Buffer Overflow\Format String Attack Version:1

Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.2 - Buffer overflows
NIST SP 800-53: SI-10 Information Input Validation (P1)
OWASP Top 10 2017: A1-Injection

Description
Format String Attack\Path 1:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=58
Status	New

Method parse_syslog_msg at line 274 of bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c receives the "<%3d>%*s %*d %*d:%*d:%*d%n %n" value from user input. This value is then used to construct a "format string" "<%3d>%*s %*d %*d:%*d:%*d%n %n", which is provided as an argument to a string formatting function in parse_syslog_msg method of bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c at line 274.

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-	bminor@@glibc-glibc-2.37.9000-CVE-

	2023-6246-FP.c	2023-6246-FP.c
Line	285	285
Object	"<%3d>%*s %*d %*d:%*d:%*d%n %n"	"<%3d>%*s %*d %*d:%*d:%*d%n %n"

Code Snippet
File Name bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Method parse_syslog_msg (const char *msg)

```
....
285.    int n = sscanf (msg, "<%3d>%*s %*d %*d:%*d:%*d%n %n"
STRINPUT(IDENT_LENGTH)
```

Format String Attack\Path 2:

Severity High
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=59
Status New

Method parse_syslog_msg at line 274 of bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c receives the "<%3d>%*s %*d %*d:%*d:%*d%n %n" value from user input. This value is then used to construct a "format string" "<%3d>%*s %*d %*d:%*d:%*d%n %n", which is provided as an argument to a string formatting function in parse_syslog_msg method of bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c at line 274.

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Line	285	285
Object	"<%3d>%*s %*d %*d:%*d:%*d%n %n"	"<%3d>%*s %*d %*d:%*d:%*d%n %n"

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Method parse_syslog_msg (const char *msg)

```
....
285.    int n = sscanf (msg, "<%3d>%*s %*d %*d:%*d:%*d%n %n"
STRINPUT(IDENT_LENGTH)
```

Format String Attack\Path 3:

Severity High
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=60
Status New

Method parse_syslog_msg at line 274 of bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c receives the "<%3d>%*s %*d %*d:%*d:%*d%n %n" value from user input. This value is then used to construct a "format string" "<%3d>%*s %*d %*d:%*d:%*d%n %n", which is provided as an argument to a string formatting function in parse_syslog_msg method of bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c at line 274.

	Source	Destination
File	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Line	285	285
Object	"<%3d>%*s %*d %*d:%*d:%*d%n %n"	"<%3d>%*s %*d %*d:%*d:%*d%n %n"

Code Snippet
File Name bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Method parse_syslog_msg (const char *msg)

```
....
285.    int n = sscanf (msg, "<%3d>%*s %*d %*d:%*d:%*d%n %n"
STRINPUT(IDENT_LENGTH)
```

Format String Attack\Path 4:

Severity High
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=61
Status New

Method parse_syslog_msg at line 274 of bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c receives the "<%3d>%*s %*d %*d:%*d:%*d%n %n" value from user input. This value is then used to construct a "format string" "<%3d>%*s %*d %*d:%*d:%*d%n %n", which is provided as an argument to a string formatting function in parse_syslog_msg method of bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c at line 274.

	Source	Destination
File	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Line	285	285
Object	"<%3d>%*s %*d %*d:%*d:%*d%n %n"	"<%3d>%*s %*d %*d:%*d:%*d%n %n"

Code Snippet
File Name bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Method parse_syslog_msg (const char *msg)

```
....
285.    int n = sscanf (msg, "<%3d>%*s %*d %*d:%*d:%*d%n %n"
STRINPUT(IDENT_LENGTH)
```

String Termination Error

Query Path:
CPP\Cx\CPP Buffer Overflow\String Termination Error Version:0

Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.2 - Buffer overflows
NIST SP 800-53: SI-10 Information Input Validation (P1)
OWASP Top 10 2017: A1-Injection

Description

String Termination Error\Path 1:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=279
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	53	92
Object	path_buf	strlen

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method get_exe_path(char *path_buf, unsigned path_buf_size)

```
....
53.      ssize_t size = readlink("/proc/self/exe", path_buf,
path_buf_size - 1);
```

File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c

Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
92.      memcpy(enclave_path + strlen(enclave_path), ENCLAVE_FILENAME,
enc_file_len);
```

String Termination Error\Path 2:

Severity	High
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=280
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-

	FP.c	FP.c
Line	53	92
Object	path_buf	strlen

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method get_exe_path(char *path_buf, unsigned path_buf_size)

```
....
53.      ssize_t size = readlink("/proc/self/exe", path_buf,
path_buf_size - 1);
```

File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
92.      memcpy(enclave_path + strlen(enclave_path), ENCLAVE_FILENAME,
enc_file_len);
```

Dangerous Functions

Query Path:
CPP\Cx\CPP Medium Threat\Dangerous Functions Version:1

Categories

OWASP Top 10 2013: A9-Using Components with Known Vulnerabilities
OWASP Top 10 2017: A9-Using Components with Known Vulnerabilities

Description
Dangerous Functions\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=289
Status	New

The dangerous function, alloca, was found in use at line 240 in chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Line	244	244
Object	alloca	alloca

Code Snippet
File Name chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c

Method	ssize_t TtyOutputHandler(const char* data, size_t count, void* user_data) {

```
....
244.    char* message = alloca(tty_prefix_len + count + 1);
```

Dangerous Functions\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=290
Status	New

The dangerous function, alloca, was found in use at line 361 in chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Line	369	369
Object	alloca	alloca

Code Snippet	
File Name	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Method	void ExitHandshake(int status, void* user_data) {

```
....
369.    char* message = alloca(message_len);
```

Dangerous Functions\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=291
Status	New

The dangerous function, alloca, was found in use at line 240 in chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Line	244	244
Object	alloca	alloca

Code Snippet

File Name	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Method	ssize_t TtyOutputHandler(const char* data, size_t count, void* user_data) {

```
....
244.    char* message = alloca(tty_prefix_len + count + 1);
```

Dangerous Functions\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=292
Status	New

The dangerous function, alloca, was found in use at line 361 in chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Line	369	369
Object	alloca	alloca

Code Snippet

File Name	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Method	void ExitHandshake(int status, void* user_data) {

```
....
369.    char* message = alloca(message_len);
```

Dangerous Functions\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=293
Status	New

The dangerous function, alloca, was found in use at line 240 in chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Line	244	244
Object	alloca	alloca

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Method ssize_t TtyOutputHandler(const char* data, size_t count, void* user_data) {

```
....
244.    char* message = alloca(tty_prefix_len + count + 1);
```

Dangerous Functions\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=294
Status	New

The dangerous function, alloca, was found in use at line 361 in chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Line	369	369
Object	alloca	alloca

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Method void ExitHandshake(int status, void* user_data) {

```
....
369.    char* message = alloca(message_len);
```

Dangerous Functions\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=295
Status	New

The dangerous function, alloca, was found in use at line 240 in chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Line	244	244
Object	alloca	alloca

Code Snippet
File Name chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Method ssize_t TtyOutputHandler(const char* data, size_t count, void* user_data) {

```
....
244.    char* message = alloca(tty_prefix_len + count + 1);
```

Dangerous Functions\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=296
Status	New

The dangerous function, alloca, was found in use at line 361 in chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Line	369	369
Object	alloca	alloca

Code Snippet
File Name chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Method void ExitHandshake(int status, void* user_data) {

```
....
369.    char* message = alloca(message_len);
```

Dangerous Functions\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=297
Status	New

The dangerous function, alloca, was found in use at line 240 in chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Line	244	244

Object	alloca	alloca

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Method ssize_t TtyOutputHandler(const char* data, size_t count, void* user_data) {

```
....
244.    char* message = alloca(tty_prefix_len + count + 1);
```

Dangerous Functions\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=298
Status	New

The dangerous function, alloca, was found in use at line 361 in chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Line	369	369
Object	alloca	alloca

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Method void ExitHandshake(int status, void* user_data) {

```
....
369.    char* message = alloca(message_len);
```

Dangerous Functions\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=299
Status	New

The dangerous function, memcpy, was found in use at line 227 in bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c	bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c

Line	356	356
Object	memcpy	memcpy

Code Snippet
File Name bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c
Method bfd_get_full_section_contents (bfd *abfd, sec_ptr sec, bfd_byte **ptr)

```
....
356.        memcpy (p, sec->contents, sz);
```

Dangerous Functions\Path 12:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=300
Status	New

The dangerous function, memcpy, was found in use at line 75 in bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	92	92
Object	memcpy	memcpy

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
92.      memcpy(enclave_path + strlen(enclave_path), ENCLAVE_FILENAME,
enc_file_len);
```

Dangerous Functions\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=301
Status	New

The dangerous function, memcpy, was found in use at line 75 in bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination

File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	92	92
Object	memcpy	memcpy

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
92.      memcpy(enclave_path + strlen(enclave_path), ENCLAVE_FILENAME,
enc_file_len);
```

Dangerous Functions\Path 14:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=302
Status New

The dangerous function, memcpy, was found in use at line 282 in c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Line	365	365
Object	memcpy	memcpy

Code Snippet
File Name c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Method static int fake_addrinfo(const char *name,

```
....
365.      memcpy(node->ai_addr, &addr.sa4, sizeof(addr.sa4));
```

Dangerous Functions\Path 15:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=303
Status New

The dangerous function, memcpy, was found in use at line 282 in c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Line	367	367
Object	memcpy	memcpy

Code Snippet
File Name c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Method static int fake_addrinfo(const char *name,

```
....
367.      memcpy(node->ai_addr, &addr.sa6, sizeof(addr.sa6));
```

Dangerous Functions\Path 16:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=304
Status New

The dangerous function, memcpy, was found in use at line 282 in c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Line	365	365
Object	memcpy	memcpy

Code Snippet
File Name c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Method static int fake_addrinfo(const char *name,

```
....
365.      memcpy(node->ai_addr, &addr.sa4, sizeof(addr.sa4));
```

Dangerous Functions\Path 17:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=305
Status New

The dangerous function, memcpy, was found in use at line 282 in c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Line	367	367
Object	memcpy	memcpy

Code Snippet
File Name c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Method static int fake_addrinfo(const char *name,

```
....
367.      memcpy(node->ai_addr, &addr.sa6, sizeof(addr.sa6));
```

Dangerous Functions\Path 18:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=306
Status New

The dangerous function, memcpy, was found in use at line 276 in c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Line	359	359
Object	memcpy	memcpy

Code Snippet
File Name c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Method static int fake_addrinfo(const char *name,

```
....
359.      memcpy(node->ai_addr, &addr.sa4, sizeof(addr.sa4));
```

Dangerous Functions\Path 19:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=307
Status New

The dangerous function, memcpy, was found in use at line 276 in c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Line	361	361
Object	memcpy	memcpy

Code Snippet
File Name c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Method static int fake_addrinfo(const char *name,

```
....
361.      memcpy(node->ai_addr, &addr.sa6, sizeof(addr.sa6));
```

Dangerous Functions\Path 20:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=308
Status	New

The dangerous function, memcpy, was found in use at line 160 in ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c	ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c
Line	175	175
Object	memcpy	memcpy

Code Snippet
File Name ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c
Method static void emitnumber(JF, double num)

```
....
175.              memcpy(x, &num, sizeof(num));
```

Dangerous Functions\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=309
Status	New

The dangerous function, memcpy, was found in use at line 182 in ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c	ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c
Line	188	188
Object	memcpy	memcpy

Code Snippet
File Name ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c
Method static void emitstring(JF, int opcode, const char *str)

```
....
188.        memcpy(x, &str, sizeof(str));
```

Dangerous Functions\Path 22:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=310
Status New

The dangerous function, memcpy, was found in use at line 2070 in cesanta@@mongoose-newest-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	cesanta@@mongoose-newest-CVE-2021-3520-FP.c	cesanta@@mongoose-newest-CVE-2021-3520-FP.c
Line	2098	2098
Object	memcpy	memcpy

Code Snippet
File Name cesanta@@mongoose-newest-CVE-2021-3520-FP.c
Method static BaseType_t prvCopyDataToQueue(Queue_t * const pxQueue, const void *pvItemToQueue, const BaseType_t xPosition)

```
....
2098.             ( void ) memcpy( ( void * ) pxQueue->pcWriteTo,
pvItemToQueue, ( size_t ) pxQueue->uxItemSize ); /*lint !e961 !e418
!e9087 MISRA exception as the casts are only redundant for some ports,
plus previous logic ensures a null pointer can only be passed to
memcpy() if the copy size is 0.  Cast to void required by function
signature and safe as no alignment requirement and copy length specified
in bytes. */
```

Dangerous Functions\Path 23:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&

	pathid=311
Status	New

The dangerous function, memcpy, was found in use at line 2070 in cesanta@@mongoose-newest-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	cesanta@@mongoose-newest-CVE-2021-3520-FP.c	cesanta@@mongoose-newest-CVE-2021-3520-FP.c
Line	2111	2111
Object	memcpy	memcpy

Code Snippet
File Name cesanta@@mongoose-newest-CVE-2021-3520-FP.c
Method static BaseType_t prvCopyDataToQueue(Queue_t * const pxQueue, const void *pvItemToQueue, const BaseType_t xPosition)

```
....
2111.             ( void ) memcpy( ( void * ) pxQueue->u.xQueue.pcReadFrom, pvItemToQueue, ( size_t ) pxQueue->uxItemSize );
/*lint !e961 !e9087 !e418 MISRA exception as the casts are only redundant for some ports.  Cast to void required by function signature and safe as no alignment requirement and copy length specified in bytes.  Assert checks null pointer only used when length is 0. */
```

Dangerous Functions\Path 24:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=312
Status	New

The dangerous function, memcpy, was found in use at line 2149 in cesanta@@mongoose-newest-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	cesanta@@mongoose-newest-CVE-2021-3520-FP.c	cesanta@@mongoose-newest-CVE-2021-3520-FP.c
Line	2162	2162
Object	memcpy	memcpy

Code Snippet
File Name cesanta@@mongoose-newest-CVE-2021-3520-FP.c
Method static void prvCopyDataFromQueue(Queue_t * const pxQueue, void * const pvBuffer)

```
....
2162.             ( void ) memcpy( ( void * ) pvBuffer, ( void * )
pxQueue->u.xQueue.pcReadFrom, ( size_t ) pxQueue->uxItemSize ); /*lint
!e961 !e418 !e9087 MISRA exception as the casts are only redundant for
some ports.  Also previous logic ensures a null pointer can only be
passed to memcpy() when the count is 0.  Cast to void required by
function signature and safe as no alignment requirement and copy length
specified in bytes. */
```

Dangerous Functions\Path 25:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=313
Status	New

The dangerous function, memcpy, was found in use at line 2446 in cesanta@@mongoose-newest-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	cesanta@@mongoose-newest-CVE-2021-3520-FP.c	cesanta@@mongoose-newest-CVE-2021-3520-FP.c
Line	2496	2496
Object	memcpy	memcpy

Code Snippet
File Name cesanta@@mongoose-newest-CVE-2021-3520-FP.c
Method BaseType_t xQueueCRReceive(QueueHandle_t xQueue, void *pvBuffer, TickType_t xTicksToWait)

```
....
2496.                         ( void ) memcpy( ( void * ) pvBuffer, (
void * ) pxQueue->u.xQueue.pcReadFrom, ( unsigned ) pxQueue->uxItemSize
);
```

Dangerous Functions\Path 26:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=314
Status	New

The dangerous function, memcpy, was found in use at line 2584 in cesanta@@mongoose-newest-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	cesanta@@mongoose-newest-CVE-	cesanta@@mongoose-newest-CVE-

	2021-3520-FP.c	2021-3520-FP.c
Line	2604	2604
Object	memcpy	memcpy

Code Snippet
File Name cesanta@@mongoose-newest-CVE-2021-3520-FP.c
Method BaseType_t xQueueCRReceiveFromISR(QueueHandle_t xQueue, void *pvBuffer, BaseType_t *pxCoRoutineWoken)

```
....
2604.                    ( void ) memcpy( ( void * ) pvBuffer, ( void * ) pxQueue->u.xQueue.pcReadFrom, ( unsigned ) pxQueue->uxItemSize );
```

Dangerous Functions\Path 27:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=315
Status	New

The dangerous function, memcpy, was found in use at line 240 in chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Line	245	245
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Method ssize_t TtyOutputHandler(const char* data, size_t count, void* user_data) {

```
....
245.    memcpy(message, s_tty_prefix, tty_prefix_len);
```

Dangerous Functions\Path 28:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=316
Status	New

The dangerous function, memcpy, was found in use at line 240 in chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Line	246	246
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Method ssize_t TtyOutputHandler(const char* data, size_t count, void* user_data) {

```
....
246.    memcpy(message + tty_prefix_len, data, count);
```

Dangerous Functions\Path 29:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=317
Status New

The dangerous function, memcpy, was found in use at line 832 in chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	894	894
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
894.            memcpy (&ctxt->ftpAddr, tmp->ai_addr, tmp->ai_addrlen);
```

Dangerous Functions\Path 30:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=318
Status New

The dangerous function, memcpy, was found in use at line 832 in chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	899	899
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
899.            memcpy (&ctxt->ftpAddr, tmp->ai_addr, tmp->ai_addrlen);
```

Dangerous Functions\Path 31:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=319
Status New

The dangerous function, memcpy, was found in use at line 832 in chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	927	927
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
927.        memcpy (&((struct sockaddr_in *)&ctxt->ftpAddr)->sin_addr,
```

Dangerous Functions\Path 32:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=320
Status New

The dangerous function, memcpy, was found in use at line 1356 in chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	1433	1433
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPGetConnection(void *ctx) {

```
....
1433.           memcpy (&((struct sockaddr_in6 *)&dataAddr)->sin6_addr,
&((struct sockaddr_in6 *)&ctxt->ftpAddr)->sin6_addr, sizeof(struct
in6_addr));
```

Dangerous Functions\Path 33:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=321
Status New

The dangerous function, memcpy, was found in use at line 1356 in chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	1449	1449
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPGetConnection(void *ctx) {

```
....
1449.           memcpy (&((struct sockaddr_in *)&dataAddr)->sin_addr,
&ad[0], 4);
```

Dangerous Functions\Path 34:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=322
Status New

The dangerous function, memcpy, was found in use at line 1356 in chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	1450	1450
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPGetConnection(void *ctx) {

```
....
1450.           memcpy (&((struct sockaddr_in *)&dataAddr)->sin_port,
&ad[4], 2);
```

Dangerous Functions\Path 35:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=323
Status New

The dangerous function, memcpy, was found in use at line 240 in chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Line	245	245
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Method ssize_t TtyOutputHandler(const char* data, size_t count, void* user_data) {

```
....
245.    memcpy(message, s_tty_prefix, tty_prefix_len);
```

Dangerous Functions\Path 36:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=324

Status	New

The dangerous function, memcpy, was found in use at line 240 in chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Line	246	246
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Method ssize_t TtyOutputHandler(const char* data, size_t count, void* user_data) {

```
....
246.    memcpy(message + tty_prefix_len, data, count);
```

Dangerous Functions\Path 37:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=325
Status	New

The dangerous function, memcpy, was found in use at line 832 in chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	894	894
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
894.            memcpy (&ctxt->ftpAddr, tmp->ai_addr, tmp->ai_addrlen);
```

Dangerous Functions\Path 38:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&

	pathid=326
Status	New

The dangerous function, memcpy, was found in use at line 832 in chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	899	899
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
899.            memcpy (&ctxt->ftpAddr, tmp->ai_addr, tmp->ai_addrlen);
```

Dangerous Functions\Path 39:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=327
Status	New

The dangerous function, memcpy, was found in use at line 832 in chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	927	927
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
927.        memcpy (&((struct sockaddr_in *)&ctxt->ftpAddr)->sin_addr,
```

Dangerous Functions\Path 40:

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

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=328
Status	New

The dangerous function, memcpy, was found in use at line 1356 in chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	1433	1433
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPGetConnection(void *ctx) {

```
....
1433.           memcpy (&((struct sockaddr_in6 *)&dataAddr)->sin6_addr,
&((struct sockaddr_in6 *)&ctxt->ftpAddr)->sin6_addr, sizeof(struct
in6_addr));
```

Dangerous Functions\Path 41:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=329
Status	New

The dangerous function, memcpy, was found in use at line 1356 in chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	1449	1449
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPGetConnection(void *ctx) {

```
....
1449.           memcpy (&((struct sockaddr_in *)&dataAddr)->sin_addr,
&ad[0], 4);
```

Dangerous Functions\Path 42:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=330
Status	New

The dangerous function, memcpy, was found in use at line 1356 in chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	1450	1450
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPGetConnection(void *ctx) {

```
....
1450.           memcpy (&((struct sockaddr_in *)&dataAddr)->sin_port,
&ad[4], 2);
```

Dangerous Functions\Path 43:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=331
Status	New

The dangerous function, memcpy, was found in use at line 240 in chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Line	245	245
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Method ssize_t TtyOutputHandler(const char* data, size_t count, void* user_data) {

```
....
245.    memcpy(message, s_tty_prefix, tty_prefix_len);
```

Dangerous Functions\Path 44:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=332
Status	New

The dangerous function, memcpy, was found in use at line 240 in chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Line	246	246
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Method ssize_t TtyOutputHandler(const char* data, size_t count, void* user_data) {

```
....
246.    memcpy(message + tty_prefix_len, data, count);
```

Dangerous Functions\Path 45:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=333
Status	New

The dangerous function, memcpy, was found in use at line 240 in chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Line	245	245
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c

Method	ssize_t TtyOutputHandler(const char* data, size_t count, void* user_data) {

```
....
245.    memcpy(message, s_tty_prefix, tty_prefix_len);
```

Dangerous Functions\Path 46:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=334
Status	New

The dangerous function, memcpy, was found in use at line 240 in chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Line	246	246
Object	memcpy	memcpy

Code Snippet	
File Name	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Method	ssize_t TtyOutputHandler(const char* data, size_t count, void* user_data) {

```
....
246.    memcpy(message + tty_prefix_len, data, count);
```

Dangerous Functions\Path 47:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=335
Status	New

The dangerous function, memcpy, was found in use at line 771 in chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	833	833
Object	memcpy	memcpy

Code Snippet

File Name	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method	xmlNanoFTPConnect(void *ctx) {

```
....
833.            memcpy (&ctxt->ftpAddr, tmp->ai_addr, tmp->ai_addrlen);
```

Dangerous Functions\Path 48:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=336
Status	New

The dangerous function, memcpy, was found in use at line 771 in chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	838	838
Object	memcpy	memcpy

Code Snippet

File Name	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method	xmlNanoFTPConnect(void *ctx) {

```
....
838.            memcpy (&ctxt->ftpAddr, tmp->ai_addr, tmp->ai_addrlen);
```

Dangerous Functions\Path 49:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=337
Status	New

The dangerous function, memcpy, was found in use at line 771 in chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	866	866
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
866.        memcpy (&((struct sockaddr_in *)&ctxt->ftpAddr)->sin_addr,
```

Dangerous Functions\Path 50:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=338
Status	New

The dangerous function, memcpy, was found in use at line 1274 in chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	1348	1348
Object	memcpy	memcpy

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPGetConnection(void *ctx) {

```
....
1348.           memcpy (&((struct sockaddr_in6 *)&dataAddr)->sin6_addr,
&((struct sockaddr_in6 *)&ctxt->ftpAddr)->sin6_addr, sizeof(struct
in6_addr));
```

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=1000009&projectid=5&pathid=858
Status	New

The variable declared in buf at bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c in line 120 is not initialized when it is used by buf at bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c in line 120.

	Source	Destination
File	bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c
Line	125	231
Object	buf	buf

Code Snippet
File Name bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
125.    char *buf = NULL;
....
231.                "\n" + (buf[bufsize - 1] == '\n'));
```

Use of Zero Initialized Pointer\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=859
Status	New

The variable declared in buf at bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c in line 120 is not initialized when it is used by buf at bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c in line 120.

	Source	Destination
File	bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c
Line	125	230
Object	buf	buf

Code Snippet
File Name bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
125.    char *buf = NULL;
....
230.      __dprintf (STDERR_FILENO, "%s%s", buf + msgoff,
```

Use of Zero Initialized Pointer\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=860
Status	New

The variable declared in buf at bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c in line 120 is not initialized when it is used by buf at bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c in line 120.

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Line	125	235
Object	buf	buf

Code Snippet
File Name bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
125.    char *buf = NULL;
....
235.                "\n" + (buf[bufsize - 1] == '\n'));
```

Use of Zero Initialized Pointer\Path 4:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=861
Status New

The variable declared in buf at bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c in line 120 is not initialized when it is used by buf at bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c in line 120.

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Line	125	234
Object	buf	buf

Code Snippet
File Name bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
125.    char *buf = NULL;
....
234.      __dprintf (STDERR_FILENO, "%s%s", buf + msgoff,
```

Use of Zero Initialized Pointer\Path 5:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=862

Status	New

The variable declared in buf at bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c in line 122 is not initialized when it is used by buf at bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c in line 122.

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Line	127	237
Object	buf	buf

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
127.    char *buf = NULL;
....
237.                "\n" + (buf[bufsize - 1] == '\n'));
```

Use of Zero Initialized Pointer\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=863
Status	New

The variable declared in buf at bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c in line 122 is not initialized when it is used by buf at bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c in line 122.

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Line	127	236
Object	buf	buf

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
127.    char *buf = NULL;
....
236.      __dprintf (STDERR_FILENO, "%s%s", buf + msgoff,
```

Use of Zero Initialized Pointer\Path 7:

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

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=864
Status	New

The variable declared in res at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c in line 82 is not initialized when it is used by res at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c in line 82.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Line	84	92
Object	res	res

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c

Method split_string(char *str, int *count)

```
....
84.      char **res = NULL;
....
92.          res = (char **)realloc(res, sizeof(char *) * (uint32)(idx + 1));
```

Use of Zero Initialized Pointer\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=865
Status	New

The variable declared in linked_func at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770 is not initialized when it is used by linked_func at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	779	836
Object	linked_func	linked_func

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c

Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
779.      WASMFunction *linked_func = NULL;
....
836.      function->import_func_linked = is_built_in_module ? NULL :
linked_func;
```

Use of Zero Initialized Pointer\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=866
Status	New

The variable declared in linked_func at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770 is not initialized when it is used by linked_func at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	779	829
Object	linked_func	linked_func

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c

Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
779.      WASMFunction *linked_func = NULL;
....
829.      function->func_ptr_linked = is_built_in_module ? linked_func :
NULL;
```

Use of Zero Initialized Pointer\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=867
Status	New

The variable declared in linked_attachment at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770 is not initialized when it is used by linked_func at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770.

	Source	Destination

File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	781	808
Object	linked_attachment	linked_func

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
781.      void *linked_attachment = NULL;
....
808.          linked_func = wasm_native_resolve_symbol(sub_module_name,
```

Use of Zero Initialized Pointer\Path 11:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=868
Status New

The variable declared in linked_signature at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770 is not initialized when it is used by linked_func at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	780	808
Object	linked_signature	linked_func

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
780.      const char *linked_signature = NULL;
....
808.          linked_func = wasm_native_resolve_symbol(sub_module_name,
```

Use of Zero Initialized Pointer\Path 12:

Severity Medium
Result State To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=869
Status	New

The variable declared in import_functions at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 1404 is not initialized when it is used by import_functions at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 1404.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	1411	1565
Object	import_functions	import_functions

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c

Method load_import_section(const uint8 *buf, const uint8 *buf_end, WASMModule *module,

```
....
1411.      WASMImport *import_functions = NULL, *import_tables = NULL;
....
1565.                        import = import_functions++;
```

Use of Zero Initialized Pointer\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=870
Status	New

The variable declared in import_memories at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 1404 is not initialized when it is used by import_memories at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 1404.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	1412	1593
Object	import_memories	import_memories

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c

Method	load_import_section(const uint8 *buf, const uint8 *buf_end, WASMModule *module,

```
....
1412.      WASMImport *import_memories = NULL, *import_globals = NULL;
....
1593.                        import = import_memories++;
```

Use of Zero Initialized Pointer\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=871
Status	New

The variable declared in types at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 4540 is not initialized when it is used by types at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 4540.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	4555	4577
Object	types	types

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method	wasm_loader_emit_br_info(WASMLoaderContext *ctx, BranchBlock *frame_csp,

```
....
4555.      uint8 *types = NULL, cell;
....
4577.              cell = wasm_value_type_cell_num(types[i]);
```

Use of Zero Initialized Pointer\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=872
Status	New

The variable declared in types at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 4540 is not initialized when it is used by types at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 4540.

	Source	Destination
File	bytecodealliance@@wasm-micro-	bytecodealliance@@wasm-micro-

	runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	4555	4582
Object	types	types

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method wasm_loader_emit_br_info(WASMLoaderContext *ctx, BranchBlock *frame_csp,

```
....
4555.      uint8 *types = NULL, cell;
....
4582.              cell = wasm_value_type_cell_num(types[i]);
```

Use of Zero Initialized Pointer\Path 16:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=873
Status New

The variable declared in types at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 4540 is not initialized when it is used by types at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 4540.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	4555	4590
Object	types	types

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method wasm_loader_emit_br_info(WASMLoaderContext *ctx, BranchBlock *frame_csp,

```
....
4555.      uint8 *types = NULL, cell;
....
4590.              cell = wasm_value_type_cell_num(types[i]);
```

Use of Zero Initialized Pointer\Path 17:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=874

Status	New

The variable declared in return_types at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5026 is not initialized when it is used by return_types at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5026.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	5034	5045
Object	return_types	return_types

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method reserve_block_ret(WASMLoaderContext *loader_ctx,

```
....
5034.      uint8 *return_types = NULL;
....
5045.          uint8 cell = wasm_value_type_cell_num(return_types[0]);
```

Use of Zero Initialized Pointer\Path 18:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=875
Status	New

The variable declared in return_types at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5026 is not initialized when it is used by return_types at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5026.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	5034	5084
Object	return_types	return_types

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method reserve_block_ret(WASMLoaderContext *loader_ctx,

```
....
5034.      uint8 *return_types = NULL;
....
5084.          uint8 cells = wasm_value_type_cell_num(return_types[i]);
```

Use of Zero Initialized Pointer\Path 19:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=876
Status	New

The variable declared in return_types at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5026 is not initialized when it is used by return_types at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5026.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	5034	5124
Object	return_types	return_types

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c

Method reserve_block_ret(WASMLoaderContext *loader_ctx,

```
....
5034.      uint8 *return_types = NULL;
....
5124.              uint8 cell =
wasm_value_type_cell_num(return_types[i]);
```

Use of Zero Initialized Pointer\Path 20:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=877
Status	New

The variable declared in return_types at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5026 is not initialized when it is used by dst_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5026.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-

	52284-FP.c	52284-FP.c
Line	5034	5133
Object	return_types	dst_offsets

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method reserve_block_ret(WASMLoaderContext *loader_ctx,

```
....
5034.      uint8 *return_types = NULL;
....
5133.                  dst_offsets[j] = dynamic_offset;
```

Use of Zero Initialized Pointer\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=878
Status	New

The variable declared in src_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5026 is not initialized when it is used by src_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5026.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	5097	5100
Object	src_offsets	src_offsets

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method reserve_block_ret(WASMLoaderContext *loader_ctx,

```
....
5097.          int16 *src_offsets = NULL;
....
5100.                                                  +
sizeof(*src_offsets)
```

Use of Zero Initialized Pointer\Path 22:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=879

Status	New

The variable declared in dst_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5026 is not initialized when it is used by dst_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5026.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	5098	5101
Object	dst_offsets	dst_offsets

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method reserve_block_ret(WASMLoaderContext *loader_ctx,

```
....
5098.          uint16 *dst_offsets = NULL;
....
5101.                                                  +
sizeof(*dst_offsets));
```

Use of Zero Initialized Pointer\Path 23:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=880
Status	New

The variable declared in cells at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5561 is not initialized when it is used by cells at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5561.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	5565	5582
Object	cells	cells

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method copy_params_to_dynamic_space(WASMLoaderContext *loader_ctx, bool is_if_block,

```
....
5565.       uint8 *cells = NULL, cell;
....
5582.           size += sizeof(*cells) + sizeof(*src_offsets);
```

Use of Zero Initialized Pointer\Path 24:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=881
Status	New

The variable declared in cells at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5561 is not initialized when it is used by cells at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5561.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	5565	5577
Object	cells	cells

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c

Method copy_params_to_dynamic_space(WASMLoaderContext *loader_ctx, bool is_if_block,

```
....
5565.       uint8 *cells = NULL, cell;
....
5577.       uint64 size = (uint64)param_count * (sizeof(*cells)
```

Use of Zero Initialized Pointer\Path 25:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=882
Status	New

The variable declared in src_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5561 is not initialized when it is used by src_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5561.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-

	52284-FP.c	52284-FP.c
Line	5566	5582
Object	src_offsets	src_offsets

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method copy_params_to_dynamic_space(WASMLoaderContext *loader_ctx, bool is_if_block,

```
....
5566.      int16 *src_offsets = NULL;
....
5582.          size += sizeof(*cells) + sizeof(*src_offsets);
```

Use of Zero Initialized Pointer\Path 26:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=883
Status New

The variable declared in src_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5561 is not initialized when it is used by src_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 5561.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	5566	5578
Object	src_offsets	src_offsets

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method copy_params_to_dynamic_space(WASMLoaderContext *loader_ctx, bool is_if_block,

```
....
5566.      int16 *src_offsets = NULL;
....
5578.                                              + sizeof(*src_offsets));
```

Use of Zero Initialized Pointer\Path 27:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&

	pathid=884
Status	New

The variable declared in res at bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c in line 72 is not initialized when it is used by res at bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c in line 72.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Line	74	82
Object	res	res

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c

Method split_string(char *str, int *count)

```
....
74.      char **res = NULL;
....
82.          res = (char**) realloc(res, sizeof(char*) * (uint32)(idx +
1));
```

Use of Zero Initialized Pointer\Path 28:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=885
Status	New

The variable declared in sub_module at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1045 is not initialized when it is used by sub_module at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1045.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	1056	1109
Object	sub_module	sub_module

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c

Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1056.       WASMModule *sub_module = NULL;
....
1109.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

Use of Zero Initialized Pointer\Path 29:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=886
Status	New

The variable declared in linked_attachment at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1045 is not initialized when it is used by linked_func at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1045.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	1059	1079
Object	linked_attachment	linked_func

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c

Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1059.       void *linked_attachment = NULL;
....
1079.       linked_func = wasm_native_resolve_symbol(
```

Use of Zero Initialized Pointer\Path 30:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=887
Status	New

The variable declared in linked_signature at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1045 is not initialized when it is used by linked_func at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1045.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-

	48105-FP.c	48105-FP.c
Line	1058	1079
Object	linked_signature	linked_func

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1058.       const char *linked_signature = NULL;
....
1079.       linked_func = wasm_native_resolve_symbol(
```

Use of Zero Initialized Pointer\Path 31:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=888
Status New

The variable declared in import_functions at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1563 is not initialized when it is used by import_functions at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1563.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	1571	1700
Object	import_functions	import_functions

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method load_import_section(const uint8 *buf, const uint8 *buf_end, WASMModule *module,

```
....
1571.       WASMImport *import_functions = NULL, *import_tables = NULL;
....
1700.                       import = import_functions++;
```

Use of Zero Initialized Pointer\Path 32:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=889

Status	New

The variable declared in import_memories at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1563 is not initialized when it is used by import_memories at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1563.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	1572	1722
Object	import_memories	import_memories

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method load_import_section(const uint8 *buf, const uint8 *buf_end, WASMModule *module,

```
....
1572.      WASMImport *import_memories = NULL, *import_globals = NULL;
....
1722.                        import = import_memories++;
```

Use of Zero Initialized Pointer\Path 33:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=890
Status	New

The variable declared in types at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5265 is not initialized when it is used by types at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5265.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	5280	5302
Object	types	types

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method wasm_loader_emit_br_info(WASMLoaderContext *ctx, BranchBlock *frame_csp,

```
....
5280.      uint8 *types = NULL, cell;
....
5302.              cell = (uint8)wasm_value_type_cell_num(types[i]);
```

Use of Zero Initialized Pointer\Path 34:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=891
Status	New

The variable declared in types at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5265 is not initialized when it is used by types at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5265.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	5280	5307
Object	types	types

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c

Method wasm_loader_emit_br_info(WASMLoaderContext *ctx, BranchBlock *frame_csp,

```
....
5280.      uint8 *types = NULL, cell;
....
5307.              cell = (uint8)wasm_value_type_cell_num(types[i]);
```

Use of Zero Initialized Pointer\Path 35:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=892
Status	New

The variable declared in types at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5265 is not initialized when it is used by types at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5265.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c

Line	5280	5315
Object	types	types

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method wasm_loader_emit_br_info(WASMLoaderContext *ctx, BranchBlock *frame_csp,

```
....
5280.      uint8 *types = NULL, cell;
....
5315.              cell = (uint8)wasm_value_type_cell_num(types[i]);
```

Use of Zero Initialized Pointer\Path 36:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=893
Status New

The variable declared in return_types at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5726 is not initialized when it is used by return_types at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5726.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	5733	5744
Object	return_types	return_types

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method reserve_block_ret(WASMLoaderContext *loader_ctx, uint8 opcode,

```
....
5733.      uint8 *return_types = NULL;
....
5744.          uint8 cell =
(uint8)wasm_value_type_cell_num(return_types[0]);
```

Use of Zero Initialized Pointer\Path 37:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=894
Status New

The variable declared in return_types at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5726 is not initialized when it is used by return_types at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5726.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	5733	5784
Object	return_types	return_types

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method reserve_block_ret(WASMLoaderContext *loader_ctx, uint8 opcode,

```
....
5733.      uint8 *return_types = NULL;
....
5784.          uint8 cells =
(uint8)wasm_value_type_cell_num(return_types[i]);
```

Use of Zero Initialized Pointer\Path 38:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=895
Status New

The variable declared in return_types at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5726 is not initialized when it is used by return_types at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5726.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	5733	5825
Object	return_types	return_types

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method reserve_block_ret(WASMLoaderContext *loader_ctx, uint8 opcode,

```
....
5733.      uint8 *return_types = NULL;
....
5825.              uint8 cell =
(uint8)wasm_value_type_cell_num(return_types[i]);
```

Use of Zero Initialized Pointer\Path 39:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=896
Status	New

The variable declared in return_types at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5726 is not initialized when it is used by dst_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5726.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	5733	5834
Object	return_types	dst_offsets

Code Snippet

File Name	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method	reserve_block_ret(WASMLoaderContext *loader_ctx, uint8 opcode,

```
....
5733.      uint8 *return_types = NULL;
....
5834.                  dst_offsets[j] = dynamic_offset;
```

Use of Zero Initialized Pointer\Path 40:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=897
Status	New

The variable declared in src_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5726 is not initialized when it is used by src_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5726.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-

	48105-FP.c	48105-FP.c
Line	5797	5801
Object	src_offsets	src_offsets

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method reserve_block_ret(WASMLoaderContext *loader_ctx, uint8 opcode,

```
....
5797.           int16 *src_offsets = NULL;
....
5801.               * (sizeof(*cells) + sizeof(*src_offsets) +
sizeof(*dst_offsets));
```

Use of Zero Initialized Pointer\Path 41:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=898
Status	New

The variable declared in dst_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5726 is not initialized when it is used by dst_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 5726.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	5798	5801
Object	dst_offsets	dst_offsets

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method reserve_block_ret(WASMLoaderContext *loader_ctx, uint8 opcode,

```
....
5798.           uint16 *dst_offsets = NULL;
....
5801.               * (sizeof(*cells) + sizeof(*src_offsets) +
sizeof(*dst_offsets));
```

Use of Zero Initialized Pointer\Path 42:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&

	pathid=899
Status	New

The variable declared in cells at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 6283 is not initialized when it is used by cells at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 6283.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	6287	6303
Object	cells	cells

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c

Method copy_params_to_dynamic_space(WASMLoaderContext *loader_ctx, bool is_if_block,

```
....
6287.      uint8 *cells = NULL, cell;
....
6303.          size += sizeof(*cells) + sizeof(*src_offsets);
```

Use of Zero Initialized Pointer\Path 43:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=900
Status	New

The variable declared in cells at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 6283 is not initialized when it is used by cells at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 6283.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	6287	6299
Object	cells	cells

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c

Method copy_params_to_dynamic_space(WASMLoaderContext *loader_ctx, bool is_if_block,

```
....
6287.       uint8 *cells = NULL, cell;
....
6299.       uint64 size = (uint64)param_count * (sizeof(*cells) +
sizeof(*src_offsets));
```

Use of Zero Initialized Pointer\Path 44:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=901
Status	New

The variable declared in src_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 6283 is not initialized when it is used by src_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 6283.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	6288	6303
Object	src_offsets	src_offsets

Code Snippet

File Name	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method	copy_params_to_dynamic_space(WASMLoaderContext *loader_ctx, bool is_if_block,

```
....
6288.       int16 *src_offsets = NULL;
....
6303.           size += sizeof(*cells) + sizeof(*src_offsets);
```

Use of Zero Initialized Pointer\Path 45:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=902
Status	New

The variable declared in src_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 6283 is not initialized when it is used by src_offsets at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 6283.

	Source	Destination
File	bytecodealliance@@wasm-micro-	bytecodealliance@@wasm-micro-

	runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	6288	6299
Object	src_offsets	src_offsets

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method copy_params_to_dynamic_space(WASMLoaderContext *loader_ctx, bool is_if_block,

```
....
6288.       int16 *src_offsets = NULL;
....
6299.       uint64 size = (uint64)param_count * (sizeof(*cells) +
sizeof(*src_offsets));
```

Use of Zero Initialized Pointer\Path 46:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=903
Status	New

The variable declared in sub_module at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1045 is not initialized when it is used by sub_module at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1045.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	1056	1109
Object	sub_module	sub_module

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1056.       WASMModule *sub_module = NULL;
....
1109.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

Use of Zero Initialized Pointer\Path 47:

Severity	Medium
Result State	To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=904
Status	New

The variable declared in linked_attachment at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1045 is not initialized when it is used by linked_func at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1045.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	1059	1079
Object	linked_attachment	linked_func

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c

Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1059.       void *linked_attachment = NULL;
....
1079.       linked_func = wasm_native_resolve_symbol(
```

Use of Zero Initialized Pointer\Path 48:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=905
Status	New

The variable declared in linked_signature at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1045 is not initialized when it is used by linked_func at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1045.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	1058	1079
Object	linked_signature	linked_func

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c

Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1058.       const char *linked_signature = NULL;
....
1079.       linked_func = wasm_native_resolve_symbol(
```

Use of Zero Initialized Pointer\Path 49:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=906
Status	New

The variable declared in import_functions at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1563 is not initialized when it is used by import_functions at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1563.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	1571	1700
Object	import_functions	import_functions

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c

Method load_import_section(const uint8 *buf, const uint8 *buf_end, WASMModule *module,

```
....
1571.       WASMImport *import_functions = NULL, *import_tables = NULL;
....
1700.                       import = import_functions++;
```

Use of Zero Initialized Pointer\Path 50:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=907
Status	New

The variable declared in import_memories at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1563 is not initialized when it is used by import_memories at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1563.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-

	52284-FP.c	52284-FP.c
Line	1572	1722
Object	import_memories	import_memories

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method load_import_section(const uint8 *buf, const uint8 *buf_end, WASMModule *module,

```
....
1572.      WASMImport *import_memories = NULL, *import_globals = NULL;
....
1722.                        import = import_memories++;
```

Buffer Overflow boundcpy WrongSizeParam

Query Path:
CPP\Cx\CPP Buffer Overflow\Buffer Overflow boundcpy WrongSizeParam Version:1

Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.2 - Buffer overflows
OWASP Top 10 2017: A1-Injection

Description
Buffer Overflow boundcpy WrongSizeParam\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=102
Status	New

The size of the buffer used by fake_addrinfo in Namespace767779162, at line 282 of c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that fake_addrinfo passes to Namespace767779162, at line 282 of c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c, to overwrite the target buffer.

	Source	Destination
File	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Line	365	365
Object	Namespace767779162	Namespace767779162

Code Snippet
File Name c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Method static int fake_addrinfo(const char *name,

```
....
365.      memcpy(node->ai_addr, &addr.sa4, sizeof(addr.sa4));
```

Buffer Overflow boundcpy WrongSizeParam\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=103
Status	New

The size of the buffer used by fake_addrinfo in Namespace767779162, at line 282 of c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that fake_addrinfo passes to Namespace767779162, at line 282 of c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c, to overwrite the target buffer.

	Source	Destination
File	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Line	367	367
Object	Namespace767779162	Namespace767779162

Code Snippet
File Name c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Method static int fake_addrinfo(const char *name,

```
....
367.      memcpy(node->ai_addr, &addr.sa6, sizeof(addr.sa6));
```

Buffer Overflow boundcpy WrongSizeParam\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=104
Status	New

The size of the buffer used by fake_addrinfo in Namespace1107175490, at line 282 of c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that fake_addrinfo passes to Namespace1107175490, at line 282 of c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c, to overwrite the target buffer.

	Source	Destination
File	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Line	365	365
Object	Namespace1107175490	Namespace1107175490

Code Snippet
File Name c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Method static int fake_addrinfo(const char *name,

```
....
365.      memcpy(node->ai_addr, &addr.sa4, sizeof(addr.sa4));
```

Buffer Overflow boundcpy WrongSizeParam\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=105
Status	New

The size of the buffer used by fake_addrinfo in Namespace1107175490, at line 282 of c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that fake_addrinfo passes to Namespace1107175490, at line 282 of c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c, to overwrite the target buffer.

	Source	Destination
File	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Line	367	367
Object	Namespace1107175490	Namespace1107175490

Code Snippet
File Name c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Method static int fake_addrinfo(const char *name,

```
....
367.      memcpy(node->ai_addr, &addr.sa6, sizeof(addr.sa6));
```

Buffer Overflow boundcpy WrongSizeParam\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=106
Status	New

The size of the buffer used by fake_addrinfo in Namespace1834928215, at line 276 of c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that fake_addrinfo passes to Namespace1834928215, at line 276 of c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c, to overwrite the target buffer.

	Source	Destination
File	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Line	359	359
Object	Namespace1834928215	Namespace1834928215

Code Snippet
File Name c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Method static int fake_addrinfo(const char *name,

```
....
359.      memcpy(node->ai_addr, &addr.sa4, sizeof(addr.sa4));
```

Buffer Overflow boundcpy WrongSizeParam\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=107
Status	New

The size of the buffer used by fake_addrinfo in Namespace1834928215, at line 276 of c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that fake_addrinfo passes to Namespace1834928215, at line 276 of c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c, to overwrite the target buffer.

	Source	Destination
File	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Line	361	361
Object	Namespace1834928215	Namespace1834928215

Code Snippet
File Name c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Method static int fake_addrinfo(const char *name,

```
....
361.      memcpy(node->ai_addr, &addr.sa6, sizeof(addr.sa6));
```

Buffer Overflow boundcpy WrongSizeParam\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=108
Status	New

The size of the buffer used by emitnumber in num, at line 160 of ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that emitnumber passes to num, at line 160 of ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c	ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c
Line	175	175
Object	num	num

Code Snippet
File Name ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c
Method static void emitnumber(JF, double num)

```
....
175.              memcpy(x, &num, sizeof(num));
```

Buffer Overflow boundcpy WrongSizeParam\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=109
Status	New

The size of the buffer used by emitstring in str, at line 182 of ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that emitstring passes to str, at line 182 of ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c, to overwrite the target buffer.

	Source	Destination
File	ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c	ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c
Line	188	188
Object	str	str

Code Snippet
File Name ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c
Method static void emitstring(JF, int opcode, const char *str)

```
....
188.          memcpy(x, &str, sizeof(str));
```

Buffer Overflow boundcpy WrongSizeParam\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=110
Status	New

The size of the buffer used by xmlNanoFTPGetConnection in in6_addr, at line 1356 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPGetConnection passes to in6_addr, at line 1356 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	1433	1433
Object	in6_addr	in6_addr

Code Snippet

File Name	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method	xmlNanoFTPGetConnection(void *ctx) {

```
....
1433.           memcpy (&((struct sockaddr_in6 *)&dataAddr)->sin6_addr,
&((struct sockaddr_in6 *)&ctxt->ftpAddr)->sin6_addr, sizeof(struct
in6_addr));
```

Buffer Overflow boundcpy WrongSizeParam\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=111
Status	New

The size of the buffer used by xmlNanoFTPGetConnection in in6_addr, at line 1356 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPGetConnection passes to in6_addr, at line 1356 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	1433	1433
Object	in6_addr	in6_addr

Code Snippet

File Name	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method	xmlNanoFTPGetConnection(void *ctx) {

```
....
1433.           memcpy (&((struct sockaddr_in6 *)&dataAddr)->sin6_addr,
&((struct sockaddr_in6 *)&ctxt->ftpAddr)->sin6_addr, sizeof(struct
in6_addr));
```

Buffer Overflow boundcpy WrongSizeParam\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=112
Status	New

The size of the buffer used by xmlNanoFTPGetConnection in in6_addr, at line 1274 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPGetConnection passes to in6_addr, at line 1274 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-	chromium@@chromium-119.0.6045.17-

	CVE-2021-3520-FP.c	CVE-2021-3520-FP.c
Line	1348	1348
Object	in6_addr	in6_addr

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPGetConnection(void *ctx) {

```
....
1348.           memcpy (&((struct sockaddr_in6 *)&dataAddr)->sin6_addr,
&((struct sockaddr_in6 *)&ctxt->ftpAddr)->sin6_addr, sizeof(struct
in6_addr));
```

Buffer Overflow boundcpy WrongSizeParam\Path 12:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=113
Status	New

The size of the buffer used by main in RuntimeInitArgs, at line 218 of bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to RuntimeInitArgs, at line 218 of bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Line	328	328
Object	RuntimeInitArgs	RuntimeInitArgs

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
328.      memset(&init_args, 0, sizeof(RuntimeInitArgs));
```

Buffer Overflow boundcpy WrongSizeParam\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=114
Status	New

The size of the buffer used by wasm_loader_ctx_init in WASMLoaderContext, at line 4005 of bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_ctx_init passes to WASMLoaderContext, at line 4005 of bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	4011	4011
Object	WASMLoaderContext	WASMLoaderContext

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method wasm_loader_ctx_init(WASMFunction *func)

```
....
4011.       memset(loader_ctx, 0, sizeof(WASMLoaderContext));
```

Buffer Overflow boundcpy WrongSizeParam\Path 14:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=115
Status New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 4156 of bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 4156 of bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	4161	4161
Object	BranchBlock	BranchBlock

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 label_type,

```
....
4161.       memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=116
Status	New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 4785 of bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 4785 of bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	4790	4790
Object	BranchBlock	BranchBlock

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c

Method wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 label_type,

```
....
4790.       memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 16:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=117
Status	New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 4785 of bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 4785 of bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	4790	4790
Object	BranchBlock	BranchBlock

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c

Method	wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 label_type,

```
....
4790.      memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 17:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=118
Status	New

The size of the buffer used by main in RuntimeInitArgs, at line 152 of bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to RuntimeInitArgs, at line 152 of bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c
Line	245	245
Object	RuntimeInitArgs	RuntimeInitArgs

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c
Method	int main(int argc, char *argv[])

```
....
245.      memset(&init_args, 0, sizeof(RuntimeInitArgs));
```

Buffer Overflow boundcpy WrongSizeParam\Path 18:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=119
Status	New

The size of the buffer used by wasm_loader_ctx_init in WASMLoaderContext, at line 3565 of bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_ctx_init passes to WASMLoaderContext, at line 3565 of bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c

Line	3571	3571
Object	WASMLoaderContext	WASMLoaderContext

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Method wasm_loader_ctx_init(WASMFunction *func)

```
....
3571.       memset(loader_ctx, 0, sizeof(WASMLoaderContext));
```

Buffer Overflow boundcpy WrongSizeParam\Path 19:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=120
Status New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 3690 of bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 3690 of bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	3695	3695
Object	BranchBlock	BranchBlock

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Method wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 type,

```
....
3695.       memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 20:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=121
Status New

The size of the buffer used by wasm_loader_ctx_init in WASMLoaderContext, at line 4451 of bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that

wasm_loader_ctx_init passes to WASMLoaderContext, at line 4451 of bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	4457	4457
Object	WASMLoaderContext	WASMLoaderContext

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Method wasm_loader_ctx_init(WASMFunction *func)

```
....
4457.       memset(loader_ctx, 0, sizeof(WASMLoaderContext));
```

Buffer Overflow boundcpy WrongSizeParam\Path 21:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=122
Status New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 4596 of bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 4596 of bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	4601	4601
Object	BranchBlock	BranchBlock

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Method wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 label_type,

```
....
4601.       memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 22:

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

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=123
Status	New

The size of the buffer used by main in RuntimeInitArgs, at line 218 of bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to RuntimeInitArgs, at line 218 of bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c
Line	328	328
Object	RuntimeInitArgs	RuntimeInitArgs

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c

Method main(int argc, char *argv[])

```
....
328.      memset(&init_args, 0, sizeof(RuntimeInitArgs));
```

Buffer Overflow boundcpy WrongSizeParam\Path 23:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=124
Status	New

The size of the buffer used by wasm_loader_ctx_init in WASMLoaderContext, at line 3883 of bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_ctx_init passes to WASMLoaderContext, at line 3883 of bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Line	3889	3889
Object	WASMLoaderContext	WASMLoaderContext

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c

Method wasm_loader_ctx_init(WASMFunction *func)

```
....
3889.       memset(loader_ctx, 0, sizeof(WASMLoaderContext));
```

Buffer Overflow boundcpy WrongSizeParam\Path 24:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=125
Status	New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 4008 of bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 4008 of bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Line	4013	4013
Object	BranchBlock	BranchBlock

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c

Method wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 label_type,

```
....
4013.       memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 25:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=126
Status	New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 4933 of bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 4933 of bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c

Line	4938	4938
Object	BranchBlock	BranchBlock

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Method wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 label_type,

```
....
4938.       memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 26:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=127
Status	New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 4933 of bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 4933 of bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Line	4938	4938
Object	BranchBlock	BranchBlock

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Method wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 label_type,

```
....
4938.       memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 27:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=128
Status	New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 5147 of bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 5147 of bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Line	5152	5152
Object	BranchBlock	BranchBlock

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Method wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 label_type,

```
....
5152.       memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 28:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=129
Status New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 5147 of bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 5147 of bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Line	5152	5152
Object	BranchBlock	BranchBlock

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Method wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 label_type,

```
....
5152.       memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 29:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=130
Status New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 5494 of bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 5494 of bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Line	5499	5499
Object	BranchBlock	BranchBlock

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Method wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 label_type,

```
....
5499.       memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 30:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=131
Status New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 5494 of bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 5494 of bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c
Line	5499	5499
Object	BranchBlock	BranchBlock

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c
Method wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 label_type,

```
....
5499.       memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 31:

Severity Medium
Result State To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=132
Status	New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 5495 of bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 5495 of bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c
Line	5500	5500
Object	BranchBlock	BranchBlock

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c
Method wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 label_type,

```
....
5500.      memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 32:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=133
Status	New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 5495 of bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 5495 of bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c
Line	5500	5500
Object	BranchBlock	BranchBlock

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c
Method wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 label_type,

```
....
5500.       memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 33:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=134
Status	New

The size of the buffer used by enclave_init in sgx_launch_token_t, at line 75 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that enclave_init passes to sgx_launch_token_t, at line 75 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	125	125
Object	sgx_launch_token_t	sgx_launch_token_t

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
125.              memset(&token, 0x0, sizeof(sgx_launch_token_t));
```

Buffer Overflow boundcpy WrongSizeParam\Path 34:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=135
Status	New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 4732 of bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 4732 of bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c
Line	4737	4737

Object	BranchBlock	BranchBlock

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c
Method wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 label_type,

```
....
4737.       memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 35:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=136
Status New

The size of the buffer used by wasm_loader_push_frame_csp in BranchBlock, at line 4732 of bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_push_frame_csp passes to BranchBlock, at line 4732 of bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c
Line	4737	4737
Object	BranchBlock	BranchBlock

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c
Method wasm_loader_push_frame_csp(WASMLoaderContext *ctx, uint8 label_type,

```
....
4737.       memset(ctx->frame_csp, 0, sizeof(BranchBlock));
```

Buffer Overflow boundcpy WrongSizeParam\Path 36:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=137
Status New

The size of the buffer used by enclave_init in sgx_launch_token_t, at line 75 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that enclave_init passes to

sgx_launch_token_t, at line 75 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	125	125
Object	sgx_launch_token_t	sgx_launch_token_t

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
125.              memset(&token, 0x0, sizeof(sgx_launch_token_t));
```

Buffer Overflow boundcpy WrongSizeParam\Path 37:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=138
Status New

The size of the buffer used by xmlNanoFTPNewCtxt in xmlNanoFTPCtxt, at line 447 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPNewCtxt passes to xmlNanoFTPCtxt, at line 447 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	457	457
Object	xmlNanoFTPCtxt	xmlNanoFTPCtxt

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPNewCtxt(const char *URL) {

```
....
457.      memset(ret, 0, sizeof(xmlNanoFTPCtxt));
```

Buffer Overflow boundcpy WrongSizeParam\Path 38:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=139
Status New

The size of the buffer used by xmlNanoFTPConnect in ->, at line 832 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPConnect passes to ->, at line 832 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	855	855
Object	->	->

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
855.      memset (&ctxt->ftpAddr, 0, sizeof(ctxt->ftpAddr));
```

Buffer Overflow boundcpy WrongSizeParam\Path 39:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=140
Status New

The size of the buffer used by xmlNanoFTPNewCtxt in xmlNanoFTPCtxt, at line 447 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPNewCtxt passes to xmlNanoFTPCtxt, at line 447 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	457	457
Object	xmlNanoFTPCtxt	xmlNanoFTPCtxt

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPNewCtxt(const char *URL) {

```
....
457.      memset(ret, 0, sizeof(xmlNanoFTPCtxt));
```

Buffer Overflow boundcpy WrongSizeParam\Path 40:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&

	pathid=141
Status	New

The size of the buffer used by xmlNanoFTPConnect in ->, at line 832 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPConnect passes to ->, at line 832 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	855	855
Object	->	->

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
855.      memset (&ctxt->ftpAddr, 0, sizeof(ctxt->ftpAddr));
```

Buffer Overflow boundcpy WrongSizeParam\Path 41:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=142
Status	New

The size of the buffer used by xmlNanoFTPNewCtxt in xmlNanoFTPCtxt, at line 430 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPNewCtxt passes to xmlNanoFTPCtxt, at line 430 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	440	440
Object	xmlNanoFTPCtxt	xmlNanoFTPCtxt

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPNewCtxt(const char *URL) {

```
....
440.      memset(ret, 0, sizeof(xmlNanoFTPCtxt));
```

Buffer Overflow boundcpy WrongSizeParam\Path 42:

Severity	Medium
Result State	To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=143
Status	New

The size of the buffer used by xmlNanoFTPConnect in ->, at line 771 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that xmlNanoFTPConnect passes to ->, at line 771 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	794	794
Object	->	->

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
794.      memset (&ctxt->ftpAddr, 0, sizeof(ctxt->ftpAddr));
```

Buffer Overflow boundcpy WrongSizeParam\Path 43:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=144
Status	New

The size of the buffer used by wasm_loader_find_block_addr in BlockAddr, at line 3223 of bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_find_block_addr passes to BlockAddr, at line 3223 of bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	3311	3311
Object	BlockAddr	BlockAddr

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method wasm_loader_find_block_addr(BlockAddr *block_addr_cache,

```
....
3311.
sizeof(BlockAddr));
```

Buffer Overflow boundcpy WrongSizeParam\Path 44:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=145
Status	New

The size of the buffer used by wasm_loader_find_block_addr in BlockAddr, at line 3756 of bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_find_block_addr passes to BlockAddr, at line 3756 of bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	3851	3851
Object	BlockAddr	BlockAddr

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c

Method wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
3851.                                                    * sizeof(BlockAddr));
```

Buffer Overflow boundcpy WrongSizeParam\Path 45:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=146
Status	New

The size of the buffer used by wasm_loader_find_block_addr in BlockAddr, at line 3756 of bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_find_block_addr passes to BlockAddr, at line 3756 of bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-

	52284-FP.c	52284-FP.c
Line	3851	3851
Object	BlockAddr	BlockAddr

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
3851.                                                   * sizeof(BlockAddr));
```

Buffer Overflow boundcpy WrongSizeParam\Path 46:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=147
Status	New

The size of the buffer used by wasm_loader_find_block_addr in BlockAddr, at line 2910 of bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_find_block_addr passes to BlockAddr, at line 2910 of bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	2986	2986
Object	BlockAddr	BlockAddr

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Method wasm_loader_find_block_addr(BlockAddr *block_addr_cache,

```
....
2986.
sizeof(BlockAddr));
```

Buffer Overflow boundcpy WrongSizeParam\Path 47:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=148
Status	New

The size of the buffer used by wasm_loader_find_block_addr in BlockAddr, at line 3609 of bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_find_block_addr passes to BlockAddr, at line 3609 of bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	3697	3697
Object	BlockAddr	BlockAddr

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Method wasm_loader_find_block_addr(BlockAddr *block_addr_cache,

```
....
3697.
sizeof(BlockAddr));
```

Buffer Overflow boundcpy WrongSizeParam\Path 48:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=149
Status New

The size of the buffer used by wasm_loader_find_block_addr in BlockAddr, at line 3187 of bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_find_block_addr passes to BlockAddr, at line 3187 of bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Line	3275	3275
Object	BlockAddr	BlockAddr

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Method wasm_loader_find_block_addr(BlockAddr *block_addr_cache,

```
....
3275.
sizeof(BlockAddr));
```

Buffer Overflow boundcpy WrongSizeParam\Path 49:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=150
Status	New

The size of the buffer used by wasm_loader_find_block_addr in BlockAddr, at line 3900 of bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_find_block_addr passes to BlockAddr, at line 3900 of bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Line	3995	3995
Object	BlockAddr	BlockAddr

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c

Method wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
3995.                                                * sizeof(BlockAddr));
```

Buffer Overflow boundcpy WrongSizeParam\Path 50:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=151
Status	New

The size of the buffer used by wasm_loader_find_block_addr in BlockAddr, at line 3900 of bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that wasm_loader_find_block_addr passes to BlockAddr, at line 3900 of bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Line	3995	3995
Object	BlockAddr	BlockAddr

Code Snippet

File Name	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Method	wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
3995.                                                    * sizeof(BlockAddr));
```

Memory Leak

Query Path:
CPP\Cx\CPP Medium Threat\Memory Leak Version:1

Categories

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

Description

Memory Leak\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=736
Status	New

	Source	Destination
File	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Line	98	98
Object	si	si

Code Snippet

File Name	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Method	static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
98.     struct StartInfo* si = malloc(sizeof(struct StartInfo));
```

Memory Leak\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=737
Status	New

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Line	98	98
Object	si	si

Code Snippet	
File Name	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Method	static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
98.    struct StartInfo* si = malloc(sizeof(struct StartInfo));
```

Memory Leak\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=738
Status	New

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Line	98	98
Object	si	si

Code Snippet	
File Name	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Method	static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
98.    struct StartInfo* si = malloc(sizeof(struct StartInfo));
```

Memory Leak\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=739
Status	New

	Source	Destination
File	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Line	98	98
Object	si	si

Code Snippet	
File Name	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Method	static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
98.     struct StartInfo* si = malloc(sizeof(struct StartInfo));
```

Memory Leak\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=740
Status	New

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Line	98	98
Object	si	si

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
98.     struct StartInfo* si = malloc(sizeof(struct StartInfo));
```

Memory Leak\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=741
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Line	2376	2376
Object	inout	inout

Code Snippet
File Name CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Method lysp_stmt_parse(struct lysc_ctx *ctx, const struct lysp_stmt *stmt, void **result, struct lysp_ext_instance **exts)

```
....
2376.           *result = inout = calloc(1, sizeof *inout);
```

Memory Leak\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=742
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Line	2448	2448
Object	type	type

Code Snippet
File Name CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Method lysp_stmt_parse(struct lysc_ctx *ctx, const struct lysp_stmt *stmt, void **result, struct lysp_ext_instance **exts)

```
....
2448.          *result = type = calloc(1, sizeof *type);
```

Memory Leak\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=743
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Line	290	290
Object	when	when

Code Snippet
File Name CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Method lysp_stmt_when(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_when **when_p)

```
....
290.      when = calloc(1, sizeof *when);
```

Memory Leak\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=744

Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Line	798	798
Object	buf	buf

Code Snippet
File Name CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Method lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, const char **pat, struct lysp_ext_instance **exts)

```
....
798.      buf = malloc(strlen(*pat) + 1);
```

Memory Leak\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=745
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Line	841	841
Object	buf	buf

Code Snippet
File Name CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Method lysp_stmt_type_pattern(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_restr **patterns)

```
....
841.      buf = malloc(arg_len + 2);
```

Memory Leak\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=746
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.0-CVE-2023-	CESNET@@libyang-v2.0.0-CVE-2023-

	26917-FP.c	26917-FP.c
Line	926	926
Object	length	length

Code Snippet
File Name CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Method lysp_stmt_type(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_type *type)

```
....
926.                type->length = calloc(1, sizeof *type->length);
```

Memory Leak\Path 12:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=747
Status New

	Source	Destination
File	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Line	949	949
Object	range	range

Code Snippet
File Name CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Method lysp_stmt_type(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_type *type)

```
....
949.                type->range = calloc(1, sizeof *type->range);
```

Memory Leak\Path 13:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=748
Status New

	Source	Destination
File	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Line	2383	2383
Object	inout	inout

Code Snippet	
File Name	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Method	lysp_stmt_parse(struct lysc_ctx *ctx, const struct lysp_stmt *stmt, void **result, struct lysp_ext_instance **exts)

```
....
2383.           *result = inout = calloc(1, sizeof *inout);
```

Memory Leak\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=749
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Line	2455	2455
Object	type	type

Code Snippet	
File Name	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Method	lysp_stmt_parse(struct lysc_ctx *ctx, const struct lysp_stmt *stmt, void **result, struct lysp_ext_instance **exts)

```
....
2455.           *result = type = calloc(1, sizeof *type);
```

Memory Leak\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=750
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Line	290	290
Object	when	when

Code Snippet	
File Name	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Method	lysp_stmt_when(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_when **when_p)

```
....
290.      when = calloc(1, sizeof *when);
```

Memory Leak\Path 16:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=751
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Line	798	798
Object	buf	buf

Code Snippet
File Name CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Method lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, const char **pat, struct lysp_ext_instance **exts)

```
....
798.      buf = malloc(strlen(*pat) + 1);
```

Memory Leak\Path 17:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=752
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Line	841	841
Object	buf	buf

Code Snippet
File Name CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Method lysp_stmt_type_pattern(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_restr **patterns)

```
....
841.      buf = malloc(arg_len + 2);
```

Memory Leak\Path 18:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=753
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Line	926	926
Object	length	length

Code Snippet
File Name CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Method lysp_stmt_type(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_type *type)

```
....
926.              type->length = calloc(1, sizeof *type->length);
```

Memory Leak\Path 19:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=754
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Line	949	949
Object	range	range

Code Snippet
File Name CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Method lysp_stmt_type(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_type *type)

```
....
949.              type->range = calloc(1, sizeof *type->range);
```

Memory Leak\Path 20:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&

	pathid=755
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Line	276	276
Object	ext_val	ext_val

Code Snippet
File Name CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Method lyd_parse_set_data_flags(struct lyd_node *node, struct lyd_meta **meta, struct lyd_ctx *lydctx,

```
....
276.              ext_val = malloc(sizeof *ext_val);
```

Memory Leak\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=756
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Line	543	543
Object	when	when

Code Snippet
File Name CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Method lysp_stmt_when(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_when **when_p)

```
....
543.      when = calloc(1, sizeof *when);
```

Memory Leak\Path 22:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=757
Status	New

	Source	Destination

File	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Line	1054	1054
Object	buf	buf

Code Snippet
File Name CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Method lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, const char **pat,

```
....
1054.       buf = malloc(strlen(*pat) + 1);
```

Memory Leak\Path 23:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=758
Status New

	Source	Destination
File	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Line	1097	1097
Object	buf	buf

Code Snippet
File Name CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Method lysp_stmt_type_pattern(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_restr **patterns)

```
....
1097.       buf = malloc(arg_len + 2);
```

Memory Leak\Path 24:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=759
Status New

	Source	Destination
File	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Line	1182	1182

Object	length	length

Code Snippet
File Name CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Method lysp_stmt_type(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_type *type)

```
....
1182.               type->length = calloc(1, sizeof *type->length);
```

Memory Leak\Path 25:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=760
Status New

	Source	Destination
File	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Line	1205	1205
Object	range	range

Code Snippet
File Name CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Method lysp_stmt_type(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_type *type)

```
....
1205.               type->range = calloc(1, sizeof *type->range);
```

Memory Leak\Path 26:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=761
Status New

	Source	Destination
File	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Line	2650	2650
Object	inout	inout

Code Snippet
File Name CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c

Method	lysp_stmt_parse(struct lysc_ctx *ctx, const struct lysp_stmt *stmt, void **result, struct lysp_ext_instance **exts)

```
....
2650.          *result = inout = calloc(1, sizeof *inout);
```

Memory Leak\Path 27:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=762
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Line	2722	2722
Object	type	type

Code Snippet	
File Name	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Method	lysp_stmt_parse(struct lysc_ctx *ctx, const struct lysp_stmt *stmt, void **result, struct lysp_ext_instance **exts)

```
....
2722.          *result = type = calloc(1, sizeof *type);
```

Memory Leak\Path 28:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=763
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Line	2376	2376
Object	inout	inout

Code Snippet	
File Name	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Method	lysp_stmt_parse(struct lysc_ctx *ctx, const struct lysp_stmt *stmt, void **result, struct lysp_ext_instance **exts)

```
....
2376.          *result = inout = calloc(1, sizeof *inout);
```

Memory Leak\Path 29:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=764
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Line	2448	2448
Object	type	type

Code Snippet
File Name CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Method lysp_stmt_parse(struct lysc_ctx *ctx, const struct lysp_stmt *stmt, void **result, struct lysp_ext_instance **exts)

```
....
2448.          *result = type = calloc(1, sizeof *type);
```

Memory Leak\Path 30:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=765
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Line	290	290
Object	when	when

Code Snippet
File Name CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Method lysp_stmt_when(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_when **when_p)

```
....
290.      when = calloc(1, sizeof *when);
```

Memory Leak\Path 31:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=766
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Line	798	798
Object	buf	buf

Code Snippet
File Name CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Method lysp_stmt_type_pattern_modifier(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, const char **pat, struct lysp_ext_instance **exts)

```
....
798.      buf = malloc(strlen(*pat) + 1);
```

Memory Leak\Path 32:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=767
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Line	841	841
Object	buf	buf

Code Snippet
File Name CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Method lysp_stmt_type_pattern(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_restr **patterns)

```
....
841.      buf = malloc(arg_len + 2);
```

Memory Leak\Path 33:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&

	pathid=768
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Line	926	926
Object	length	length

Code Snippet
File Name CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Method lysp_stmt_type(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_type *type)

```
....
926.                type->length = calloc(1, sizeof *type->length);
```

Memory Leak\Path 34:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=769
Status	New

	Source	Destination
File	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Line	949	949
Object	range	range

Code Snippet
File Name CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Method lysp_stmt_type(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_type *type)

```
....
949.                type->range = calloc(1, sizeof *type->range);
```

Memory Leak\Path 35:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=770
Status	New

	Source	Destination

File	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Line	279	279
Object	ext_val	ext_val

Code Snippet
File Name CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Method lyd_parse_set_data_flags(struct lyd_node *node, struct lyd_meta **meta, struct lyd_ctx *lydctx,

```
....
279.              ext_val = malloc(sizeof *ext_val);
```

Memory Leak\Path 36:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=771
Status New

	Source	Destination
File	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Line	601	601
Object	when	when

Code Snippet
File Name CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Method lysp_stmt_when(struct lysp_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_when **when_p)

```
....
601.      when = calloc(1, sizeof *when);
```

Memory Leak\Path 37:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=772
Status New

	Source	Destination
File	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Line	1112	1112

Object	buf	buf

Code Snippet
File Name CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Method lysp_stmt_type_pattern_modifier(struct lysp_ctx *ctx, const struct lysp_stmt *stmt, const char **pat,

```
....
1112.       buf = malloc(strlen(*pat) + 1);
```

Memory Leak\Path 38:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=773
Status New

	Source	Destination
File	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Line	1155	1155
Object	buf	buf

Code Snippet
File Name CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Method lysp_stmt_type_pattern(struct lysp_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_restr **patterns)

```
....
1155.       buf = malloc(arg_len + 2);
```

Memory Leak\Path 39:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=774
Status New

	Source	Destination
File	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Line	1743	1743
Object	length	length

Code Snippet
File Name CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c

Method	lysp_stmt_type(struct lysp_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_type *type)

```
....
1743.               type->length = calloc(1, sizeof *type->length);
```

Memory Leak\Path 40:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=775
Status	New

	Source	Destination
File	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Line	1766	1766
Object	range	range

Code Snippet

File Name	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Method	lysp_stmt_type(struct lysp_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_type *type)

```
....
1766.               type->range = calloc(1, sizeof *type->range);
```

Memory Leak\Path 41:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=776
Status	New

	Source	Destination
File	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Line	3150	3150
Object	inout	inout

Code Snippet

File Name	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Method	lysp_stmt_parse(struct lysp_ctx *pctx, const struct lysp_stmt *stmt, void **result, struct lysp_ext_instance **exts)

```
....
3150.           *result = inout = calloc(1, sizeof *inout);
```

Memory Leak\Path 42:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=777
Status	New

	Source	Destination
File	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Line	3243	3243
Object	restr	restr

Code Snippet
File Name CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Method lysp_stmt_parse(struct lysp_ctx *pctx, const struct lysp_stmt *stmt, void **result, struct lysp_ext_instance **exts)

```
....
3243.           *result = restr = calloc(1, sizeof *restr);
```

Memory Leak\Path 43:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=778
Status	New

	Source	Destination
File	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Line	3278	3278
Object	mod	mod

Code Snippet
File Name CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Method lysp_stmt_parse(struct lysp_ctx *pctx, const struct lysp_stmt *stmt, void **result, struct lysp_ext_instance **exts)

```
....
3278.           *result = mod = calloc(1, sizeof *mod);
```

Memory Leak\Path 44:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=779
Status	New

	Source	Destination
File	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Line	3322	3322
Object	submod	submod

Code Snippet
File Name CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Method lysp_stmt_parse(struct lysp_ctx *pctx, const struct lysp_stmt *stmt, void **result, struct lysp_ext_instance **exts)

```
....
3322.           *result = submod = calloc(1, sizeof *submod);
```

Memory Leak\Path 45:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=780
Status	New

	Source	Destination
File	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Line	3330	3330
Object	type	type

Code Snippet
File Name CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Method lysp_stmt_parse(struct lysp_ctx *pctx, const struct lysp_stmt *stmt, void **result, struct lysp_ext_instance **exts)

```
....
3330.           *result = type = calloc(1, sizeof *type);
```

Memory Leak\Path 46:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&

	pathid=781
Status	New

	Source	Destination
File	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Line	101	101
Object	argv_	argv_

Code Snippet
File Name chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
101.    si->argv_ = calloc(argc + 1, sizeof(char*));
```

Memory Leak\Path 47:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=782
Status	New

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Line	101	101
Object	argv_	argv_

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
101.    si->argv_ = calloc(argc + 1, sizeof(char*));
```

Memory Leak\Path 48:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=783
Status	New

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c

Line	101	101
Object	argv_	argv_

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
101.    si->argv_ = calloc(argc + 1, sizeof(char*));
```

Memory Leak\Path 49:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=784
Status New

	Source	Destination
File	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Line	101	101
Object	argv_	argv_

Code Snippet
File Name chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
101.    si->argv_ = calloc(argc + 1, sizeof(char*));
```

Memory Leak\Path 50:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=785
Status New

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Line	101	101
Object	argv_	argv_

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c

Method	static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
101.    si->argv_ = calloc(argc + 1, sizeof(char*));
```

Buffer Overflow AddressOfLocalVarReturned

Query Path:
CPP\Cx\CPP Buffer Overflow\Buffer Overflow AddressOfLocalVarReturned Version:1

Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.2 - Buffer overflows
NIST SP 800-53: SC-5 Denial of Service Protection (P1)
OWASP Top 10 2017: A1-Injection

Description

Buffer Overflow AddressOfLocalVarReturned\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=72
Status	New

The pointer NewWebUI at brave@@brave-core-v1.11.4-CVE-2023-52263-TP.c in line 103 is being used after it has been freed.

	Source	Destination
File	brave@@brave-core-v1.11.4-CVE-2023-52263-TP.c	brave@@brave-core-v1.11.4-CVE-2023-52263-TP.c
Line	123	123
Object	NewWebUI	NewWebUI

Code Snippet	
File Name	brave@@brave-core-v1.11.4-CVE-2023-52263-TP.c
Method	WebUIFactoryFunction GetWebUIFactoryFunction(WebUI* web_ui,

```
....
123.      return &NewWebUI<BasicUI>;
```

Buffer Overflow AddressOfLocalVarReturned\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=73
Status	New

The pointer NewWebUI at brave@@brave-core-v1.6.34-CVE-2023-52263-TP.c in line 103 is being used after it has been freed.

	Source	Destination

File	brave@@brave-core-v1.6.34-CVE-2023-52263-TP.c	brave@@brave-core-v1.6.34-CVE-2023-52263-TP.c
Line	123	123
Object	NewWebUI	NewWebUI

Code Snippet
File Name brave@@brave-core-v1.6.34-CVE-2023-52263-TP.c
Method WebUIFactoryFunction GetWebUIFactoryFunction(WebUI* web_ui,

```
....
123.      return &NewWebUI<BasicUI>;
```

Buffer Overflow AddressOfLocalVarReturned\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=74
Status	New

The pointer p_end at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 225 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	292	292
Object	p_end	p_end

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method check_utf8_str(const uint8* str, uint32 len)

```
....
292.      return (p == p_end);
```

Buffer Overflow AddressOfLocalVarReturned\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=75
Status	New

The pointer type_str at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 199 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	204	204
Object	type_str	type_str

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method type2str(uint8 type)

```
....
204.          return type_str[type - VALUE_TYPE_V128];
```

Buffer Overflow AddressOfLocalVarReturned\Path 5:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=76
Status New

The pointer p_end at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 289 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	350	350
Object	p_end	p_end

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method check_utf8_str(const uint8 *str, uint32 len)

```
....
350.      return (p == p_end);
```

Buffer Overflow AddressOfLocalVarReturned\Path 6:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=77
Status New

The pointer type_str at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 199 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	204	204
Object	type_str	type_str

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method type2str(uint8 type)

```
....
204.          return type_str[type - VALUE_TYPE_V128];
```

Buffer Overflow AddressOfLocalVarReturned\Path 7:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=78
Status New

The pointer p_end at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 289 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	350	350
Object	p_end	p_end

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method check_utf8_str(const uint8 *str, uint32 len)

```
....
350.      return (p == p_end);
```

Buffer Overflow AddressOfLocalVarReturned\Path 8:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&

	pathid=79
Status	New

The pointer type_str at bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c in line 193 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	198	198
Object	type_str	type_str

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Method type2str(uint8 type)

```
....
198.          return type_str[type - VALUE_TYPE_V128];
```

Buffer Overflow AddressOfLocalVarReturned\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=80
Status	New

The pointer p_end at bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c in line 287 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	354	354
Object	p_end	p_end

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Method check_utf8_str(const uint8* str, uint32 len)

```
....
354.       return (p == p_end);
```

Buffer Overflow AddressOfLocalVarReturned\Path 10:

Severity	Medium

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=81
Status	New

The pointer p_end at bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c in line 251 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Line	318	318
Object	p_end	p_end

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Method check_utf8_str(const uint8* str, uint32 len)

```
....
318.       return (p == p_end);
```

Buffer Overflow AddressOfLocalVarReturned\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=82
Status	New

The pointer type_str at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c in line 203 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Line	208	208
Object	type_str	type_str

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Method type2str(uint8 type)

```
....
208.           return type_str[type - VALUE_TYPE_V128];
```

Buffer Overflow AddressOfLocalVarReturned\Path 12:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=83
Status	New

The pointer p_end at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c in line 293 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Line	354	354
Object	p_end	p_end

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Method check_utf8_str(const uint8 *str, uint32 len)

```
....
354.       return (p == p_end);
```

Buffer Overflow AddressOfLocalVarReturned\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=84
Status	New

The pointer type_str at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c in line 203 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Line	208	208
Object	type_str	type_str

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Method type2str(uint8 type)

```
....
208.           return type_str[type - VALUE_TYPE_V128];
```

Buffer Overflow AddressOfLocalVarReturned\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=85
Status	New

The pointer p_end at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c in line 293 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Line	354	354
Object	p_end	p_end

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Method check_utf8_str(const uint8 *str, uint32 len)

```
....
354.      return (p == p_end);
```

Buffer Overflow AddressOfLocalVarReturned\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=86
Status	New

The pointer type_str at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c in line 206 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Line	211	211
Object	type_str	type_str

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Method type2str(uint8 type)

```
....
211.          return type_str[type - VALUE_TYPE_V128];
```

Buffer Overflow AddressOfLocalVarReturned\Path 16:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=87
Status	New

The pointer p_end at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c in line 296 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Line	357	357
Object	p_end	p_end

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Method check_utf8_str(const uint8 *str, uint32 len)

```
....
357.      return (p == p_end);
```

Buffer Overflow AddressOfLocalVarReturned\Path 17:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=88
Status	New

The pointer type_str at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c in line 206 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Line	211	211
Object	type_str	type_str

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c

Method	type2str(uint8 type)

```
....
211.          return type_str[type - VALUE_TYPE_V128];
```

Buffer Overflow AddressOfLocalVarReturned\Path 18:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=89
Status	New

The pointer p_end at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c in line 296 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Line	357	357
Object	p_end	p_end

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Method	check_utf8_str(const uint8 *str, uint32 len)

```
....
357.      return (p == p_end);
```

Buffer Overflow AddressOfLocalVarReturned\Path 19:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=90
Status	New

The pointer type_str at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c in line 206 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Line	211	211
Object	type_str	type_str

Code Snippet

File Name	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Method	type2str(uint8 type)

```
....
211.          return type_str[type - VALUE_TYPE_V128];
```

Buffer Overflow AddressOfLocalVarReturned\Path 20:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=91
Status	New

The pointer p_end at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c in line 296 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Line	357	357
Object	p_end	p_end

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Method	check_utf8_str(const uint8 *str, uint32 len)

```
....
357.      return (p == p_end);
```

Buffer Overflow AddressOfLocalVarReturned\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=92
Status	New

The pointer type_str at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c in line 206 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c
Line	211	211
Object	type_str	type_str

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c
Method	type2str(uint8 type)

```
....
211.          return type_str[type - VALUE_TYPE_V128];
```

Buffer Overflow AddressOfLocalVarReturned\Path 22:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=93
Status	New

The pointer p_end at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c in line 296 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c
Line	357	357
Object	p_end	p_end

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c
Method	check_utf8_str(const uint8 *str, uint32 len)

```
....
357.      return (p == p_end);
```

Buffer Overflow AddressOfLocalVarReturned\Path 23:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=94
Status	New

The pointer type_str at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c in line 206 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c
Line	211	211
Object	type_str	type_str

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c
Method	type2str(uint8 type)

```
....
211.          return type_str[type - VALUE_TYPE_V128];
```

Buffer Overflow AddressOfLocalVarReturned\Path 24:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=95
Status	New

The pointer p_end at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c in line 296 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c
Line	357	357
Object	p_end	p_end

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c
Method	check_utf8_str(const uint8 *str, uint32 len)

```
....
357.      return (p == p_end);
```

Buffer Overflow AddressOfLocalVarReturned\Path 25:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=96
Status	New

The pointer type_str at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c in line 206 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c
Line	211	211

Object	type_str	type_str

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c
Method type2str(uint8 type)

```
....
211.          return type_str[type - VALUE_TYPE_V128];
```

Buffer Overflow AddressOfLocalVarReturned\Path 26:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=97
Status	New

The pointer p_end at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c in line 296 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c
Line	357	357
Object	p_end	p_end

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c
Method check_utf8_str(const uint8 *str, uint32 len)

```
....
357.      return (p == p_end);
```

Buffer Overflow AddressOfLocalVarReturned\Path 27:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=98
Status	New

The pointer type_str at bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c in line 196 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c

Line	201	201
Object	type_str	type_str

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c
Method type2str(uint8 type)

```
....
201.          return type_str[type - VALUE_TYPE_V128];
```

Buffer Overflow AddressOfLocalVarReturned\Path 28:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=99
Status New

The pointer p_end at bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c in line 286 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c
Line	347	347
Object	p_end	p_end

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c
Method check_utf8_str(const uint8 *str, uint32 len)

```
....
347.      return (p == p_end);
```

Buffer Overflow AddressOfLocalVarReturned\Path 29:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=100
Status New

The pointer type_str at bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c in line 196 is being used after it has been freed.

	Source	Destination

File	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c
Line	201	201
Object	type_str	type_str

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c
Method type2str(uint8 type)

```
....
201.          return type_str[type - VALUE_TYPE_V128];
```

Buffer Overflow AddressOfLocalVarReturned\Path 30:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=101
Status	New

The pointer p_end at bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c in line 286 is being used after it has been freed.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c
Line	347	347
Object	p_end	p_end

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c
Method check_utf8_str(const uint8 *str, uint32 len)

```
....
347.      return (p == p_end);
```

MemoryFree on StackVariable

Query Path:
CPP\Cx\CPP Medium Threat\MemoryFree on StackVariable Version:0

Description

MemoryFree on StackVariable\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&

	pathid=264
Status	New

Calling free() (line 227) on a variable that was not dynamically allocated (line 227) in file bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c may result with a crash.

	Source	Destination
File	bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c	bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c
Line	291	291
Object	p	p

Code Snippet
File Name bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c
Method bfd_get_full_section_contents (bfd *abfd, sec_ptr sec, bfd_byte **ptr)

```
....
291.            free (p);
```

MemoryFree on StackVariable\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=265
Status	New

Calling free() (line 227) on a variable that was not dynamically allocated (line 227) in file bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c may result with a crash.

	Source	Destination
File	bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c	bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c
Line	334	334
Object	p	p

Code Snippet
File Name bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c
Method bfd_get_full_section_contents (bfd *abfd, sec_ptr sec, bfd_byte **ptr)

```
....
334.            free (p);
```

MemoryFree on StackVariable\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=266

Status	New

Calling free() (line 227) on a variable that was not dynamically allocated (line 227) in file bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c may result with a crash.

	Source	Destination
File	bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c	bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c
Line	336	336
Object	compressed_buffer	compressed_buffer

Code Snippet
File Name bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c
Method bfd_get_full_section_contents (bfd *abfd, sec_ptr sec, bfd_byte **ptr)

```
....
336.          free (compressed_buffer);
```

MemoryFree on StackVariable\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=267
Status	New

Calling free() (line 227) on a variable that was not dynamically allocated (line 227) in file bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c may result with a crash.

	Source	Destination
File	bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c	bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c
Line	340	340
Object	compressed_buffer	compressed_buffer

Code Snippet
File Name bminor@@binutils-gdb-binutils-2_35_2-CVE-2023-25586-FP.c
Method bfd_get_full_section_contents (bfd *abfd, sec_ptr sec, bfd_byte **ptr)

```
....
340.        free (compressed_buffer);
```

MemoryFree on StackVariable\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=268
Status	New

Calling free() (line 117) on a variable that was not dynamically allocated (line 117) in file bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c may result with a crash.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Line	146	146
Object	cmd	cmd

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Method app_instance_repl(wasm_module_inst_t module_inst)

```
....
146.        free(cmd);
```

MemoryFree on StackVariable\Path 6:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=269
Status New

Calling free() (line 96) on a variable that was not dynamically allocated (line 96) in file bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c may result with a crash.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Line	121	121
Object	cmd	cmd

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Method app_instance_repl(wasm_module_inst_t module_inst)

```
....
121.        free(cmd);
```

MemoryFree on StackVariable\Path 7:

Severity Medium

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=270
Status	New

Calling free() (line 98) on a variable that was not dynamically allocated (line 98) in file bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c may result with a crash.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c
Line	123	123
Object	cmd	cmd

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c

Method app_instance_repl(wasm_module_inst_t module_inst)

```
....
123.        free(cmd);
```

MemoryFree on StackVariable\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=271
Status	New

Calling free() (line 117) on a variable that was not dynamically allocated (line 117) in file bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c may result with a crash.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c
Line	146	146
Object	cmd	cmd

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c

Method app_instance_repl(wasm_module_inst_t module_inst)

```
....
146.        free(cmd);
```

MemoryFree on StackVariable\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=272
Status	New

Calling free() (line 311) on a variable that was not dynamically allocated (line 311) in file bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c may result with a crash.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	340	340
Object	cmd	cmd

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method app_instance_repl(void *module_inst, int app_argc, char **app_argv)

```
....
340.        free(cmd);
```

MemoryFree on StackVariable\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=273
Status	New

Calling free() (line 311) on a variable that was not dynamically allocated (line 311) in file bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c may result with a crash.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	340	340
Object	cmd	cmd

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method app_instance_repl(void *module_inst, int app_argc, char **app_argv)

```
....
340.       free(cmd);
```

MemoryFree on StackVariable\Path 11:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=274
Status New

Calling free() (line 322) on a variable that was not dynamically allocated (line 322) in file chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c may result with a crash.

	Source	Destination
File	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Line	348	348
Object	si	si

Code Snippet
File Name chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Method void* MainThread(void* info) {

```
....
348.     free(si);
```

MemoryFree on StackVariable\Path 12:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=275
Status New

Calling free() (line 322) on a variable that was not dynamically allocated (line 322) in file chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c may result with a crash.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Line	348	348
Object	si	si

Code Snippet

File Name	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Method	void* MainThread(void* info) {

```
....
348.    free(si);
```

MemoryFree on StackVariable\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=276
Status	New

Calling free() (line 322) on a variable that was not dynamically allocated (line 322) in file chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c may result with a crash.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Line	348	348
Object	si	si

Code Snippet

File Name	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Method	void* MainThread(void* info) {

```
....
348.    free(si);
```

MemoryFree on StackVariable\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=277
Status	New

Calling free() (line 322) on a variable that was not dynamically allocated (line 322) in file chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c may result with a crash.

	Source	Destination
File	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Line	348	348
Object	si	si

Code Snippet

File Name	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c

Method	void* MainThread(void* info) {

```
....
348.    free(si);
```

MemoryFree on StackVariable\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=278
Status	New

Calling free() (line 322) on a variable that was not dynamically allocated (line 322) in file chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c may result with a crash.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Line	348	348
Object	si	si

Code Snippet	
File Name	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Method	void* MainThread(void* info) {

```
....
348.    free(si);
```

Wrong Size t Allocation

Query Path:
CPP\Cx\CPP Integer Overflow\Wrong Size t Allocation Version:0

Description

Wrong Size t Allocation\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=281
Status	New

The function pl in cesanta@@mongoose-newest-CVE-2020-8597-TP.c at line 1197 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

	Source	Destination
File	cesanta@@mongoose-newest-CVE-2020-8597-TP.c	cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Line	1211	1211

Object	pl	pl

Code Snippet
File Name cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Method name_of_pn_file()

```
....
1211.       path = malloc(pl);
```

Wrong Size t Allocation\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=282
Status	New

The function arg_len in CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c at line 830 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

	Source	Destination
File	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Line	841	841
Object	arg_len	arg_len

Code Snippet
File Name CESNET@@libyang-v2.0.0-CVE-2023-26917-FP.c
Method lysp_stmt_type_pattern(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_restr **patterns)

```
....
841.      buf = malloc(arg_len + 2);
```

Wrong Size t Allocation\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=283
Status	New

The function arg_len in CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c at line 830 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

	Source	Destination
File	CESNET@@libyang-v2.0.164-CVE-2023-	CESNET@@libyang-v2.0.164-CVE-2023-

	26917-TP.c	26917-TP.c
Line	841	841
Object	arg_len	arg_len

Code Snippet
File Name CESNET@@libyang-v2.0.164-CVE-2023-26917-TP.c
Method lysp_stmt_type_pattern(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_restr **patterns)

```
....
841.      buf = malloc(arg_len + 2);
```

Wrong Size t Allocation\Path 4:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=284
Status New

The function arg_len in CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c at line 1086 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

	Source	Destination
File	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c	CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Line	1097	1097
Object	arg_len	arg_len

Code Snippet
File Name CESNET@@libyang-v2.0.231-CVE-2023-26917-TP.c
Method lysp_stmt_type_pattern(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_restr **patterns)

```
....
1097.      buf = malloc(arg_len + 2);
```

Wrong Size t Allocation\Path 5:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=285
Status New

The function arg_len in CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c at line 830 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

	Source	Destination
File	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c	CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Line	841	841
Object	arg_len	arg_len

Code Snippet
File Name CESNET@@libyang-v2.0.88-CVE-2023-26917-FP.c
Method lysp_stmt_type_pattern(struct lys_parser_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_restr **patterns)

```
....
841.      buf = malloc(arg_len + 2);
```

Wrong Size t Allocation\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=286
Status	New

The function arg_len in CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c at line 1144 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

	Source	Destination
File	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c	CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Line	1155	1155
Object	arg_len	arg_len

Code Snippet
File Name CESNET@@libyang-v2.1.4-CVE-2023-26917-TP.c
Method lysp_stmt_type_pattern(struct lysp_ctx *ctx, const struct lysp_stmt *stmt, struct lysp_restr **patterns)

```
....
1155.      buf = malloc(arg_len + 2);
```

Wrong Size t Allocation\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=287
Status	New

The function bufsize in bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c at line 120 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Line	204	204
Object	bufsize	bufsize

Code Snippet
File Name bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
204.        buf = malloc ((bufsize + 1) * sizeof (char));
```

Wrong Size t Allocation\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=288
Status	New

The function bufsize in bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c at line 122 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Line	206	206
Object	bufsize	bufsize

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
206.        buf = malloc ((bufsize + 1) * sizeof (char));
```

Stored Buffer Overflow fgets

Query Path:
CPP\Cx\CPP Stored Vulnerabilities\Stored Buffer Overflow fgets Version:1

Categories

NIST SP 800-53: SI-10 Information Input Validation (P1)
OWASP Top 10 2017: A1-Injection

Description

Stored Buffer Overflow fgets\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1273
Status	New

The size of the buffer used by ares__read_line in bytestoread, at line 31 of c-ares@@c-ares-cares-1_16_0-CVE-2024-25629-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ares__read_line passes to BinaryExpr, at line 31 of c-ares@@c-ares-cares-1_16_0-CVE-2024-25629-TP.c, to overwrite the target buffer.

	Source	Destination
File	c-ares@@c-ares-cares-1_16_0-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_16_0-CVE-2024-25629-TP.c
Line	49	49
Object	BinaryExpr	bytestoread

Code Snippet
File Name c-ares@@c-ares-cares-1_16_0-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Stored Buffer Overflow fgets\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1274
Status	New

The size of the buffer used by ares__read_line in bytestoread, at line 31 of c-ares@@c-ares-c-ares-1_17_0-CVE-2024-25629-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ares__read_line passes to BinaryExpr, at line 31 of c-ares@@c-ares-c-ares-1_17_0-CVE-2024-25629-TP.c, to overwrite the target buffer.

	Source	Destination
File	c-ares@@c-ares-c-ares-1_17_0-CVE-2024-25629-TP.c	c-ares@@c-ares-c-ares-1_17_0-CVE-2024-25629-TP.c
Line	49	49
Object	BinaryExpr	bytestoread

Code Snippet
File Name c-ares@@c-ares-c-ares-1_17_0-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Stored Buffer Overflow fgets\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1275
Status	New

The size of the buffer used by ares__read_line in bytestoread, at line 31 of c-ares@@c-ares-cares-1_17_2-CVE-2024-25629-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ares__read_line passes to BinaryExpr, at line 31 of c-ares@@c-ares-cares-1_17_2-CVE-2024-25629-TP.c, to overwrite the target buffer.

	Source	Destination
File	c-ares@@c-ares-cares-1_17_2-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_17_2-CVE-2024-25629-TP.c
Line	49	49
Object	BinaryExpr	bytestoread

Code Snippet
File Name c-ares@@c-ares-cares-1_17_2-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Stored Buffer Overflow fgets\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1276
Status	New

The size of the buffer used by ares__read_line in bytestoread, at line 31 of c-ares@@c-ares-cares-1_18_0-CVE-2024-25629-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ares__read_line passes to BinaryExpr, at line 31 of c-ares@@c-ares-cares-1_18_0-CVE-2024-25629-TP.c, to overwrite the target buffer.

	Source	Destination
File	c-ares@@c-ares-cares-1_18_0-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_18_0-CVE-2024-25629-TP.c
Line	49	49
Object	BinaryExpr	bytestoread

Code Snippet
File Name c-ares@@c-ares-cares-1_18_0-CVE-2024-25629-TP.c

Method	int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Stored Buffer Overflow fgets\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1277
Status	New

The size of the buffer used by ares__read_line in bytestoread, at line 31 of c-ares@@c-ares-cares-1_19_0-CVE-2024-25629-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ares__read_line passes to BinaryExpr, at line 31 of c-ares@@c-ares-cares-1_19_0-CVE-2024-25629-TP.c, to overwrite the target buffer.

	Source	Destination
File	c-ares@@c-ares-cares-1_19_0-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_19_0-CVE-2024-25629-TP.c
Line	49	49
Object	BinaryExpr	bytestoread

Code Snippet

File Name	c-ares@@c-ares-cares-1_19_0-CVE-2024-25629-TP.c
Method	int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Stored Buffer Overflow fgets\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1278
Status	New

The size of the buffer used by ares__read_line in bytestoread, at line 31 of c-ares@@c-ares-cares-1_19_1-CVE-2024-25629-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ares__read_line passes to BinaryExpr, at line 31 of c-ares@@c-ares-cares-1_19_1-CVE-2024-25629-TP.c, to overwrite the target buffer.

	Source	Destination
File	c-ares@@c-ares-cares-1_19_1-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_19_1-CVE-2024-25629-TP.c
Line	49	49
Object	BinaryExpr	bytestoread

Code Snippet

File Name	c-ares@@c-ares-cares-1_19_1-CVE-2024-25629-TP.c
Method	int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Stored Buffer Overflow fgets\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1279
Status	New

The size of the buffer used by ares__read_line in bytestoread, at line 42 of c-ares@@c-ares-cares-1_20_0-CVE-2024-25629-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ares__read_line passes to BinaryExpr, at line 42 of c-ares@@c-ares-cares-1_20_0-CVE-2024-25629-TP.c, to overwrite the target buffer.

	Source	Destination
File	c-ares@@c-ares-cares-1_20_0-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_20_0-CVE-2024-25629-TP.c
Line	60	60
Object	BinaryExpr	bytestoread

Code Snippet

File Name	c-ares@@c-ares-cares-1_20_0-CVE-2024-25629-TP.c
Method	int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
60.         if (!fgets(*buf + offset, bytestoread, fp))
```

Stored Buffer Overflow fgets\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1280
Status	New

The size of the buffer used by ares__read_line in bytestoread, at line 41 of c-ares@@c-ares-cares-1_26_0-CVE-2024-25629-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ares__read_line passes to BinaryExpr, at line 41 of c-ares@@c-ares-cares-1_26_0-CVE-2024-25629-TP.c, to overwrite the target buffer.

	Source	Destination
File	c-ares@@c-ares-cares-1_26_0-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_26_0-CVE-2024-25629-TP.c
Line	58	58
Object	BinaryExpr	bytestoread

Code Snippet
File Name c-ares@@c-ares-cares-1_26_0-CVE-2024-25629-TP.c
Method ares_status_t ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
58.      if (!fgets(*buf + offset, bytestoread, fp)) {
```

Inadequate Encryption Strength

Query Path:
CPP\Cx\CPP Medium Threat\Inadequate Encryption Strength Version:1

Categories

FISMA 2014: Configuration Management
NIST SP 800-53: SC-13 Cryptographic Protection (P1)
OWASP Top 10 2017: A3-Sensitive Data Exposure

Description

Inadequate Encryption Strength\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=791
Status	New

The application uses a weak cryptographic algorithm, lwip_md5_update at line 1311 of cesanta@@mongoose-newest-CVE-2020-8597-TP.c, to protect sensitive personal information secret_len, from cesanta@@mongoose-newest-CVE-2020-8597-TP.c at line 1311.

	Source	Destination
File	cesanta@@mongoose-newest-CVE-2020-8597-TP.c	cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Line	1450	1450
Object	secret_len	lwip_md5_update

Code Snippet
File Name cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Method static void eap_request(ppp_pcb *pcb, u_char *inp, int id, int len) {

```
....
1450.              lwip_md5_update(&mdContext, (u_char *)secret,
secret_len);
```

Inadequate Encryption Strength\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=792
Status	New

The application uses a weak cryptographic algorithm, lwip_md5_update at line 1311 of cesanta@@mongoose-newest-CVE-2020-8597-TP.c, to protect sensitive personal information secret, from cesanta@@mongoose-newest-CVE-2020-8597-TP.c at line 1311.

	Source	Destination
File	cesanta@@mongoose-newest-CVE-2020-8597-TP.c	cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Line	1450	1450
Object	secret	lwip_md5_update

Code Snippet
File Name cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Method static void eap_request(ppp_pcb *pcb, u_char *inp, int id, int len) {

```
....
1450.             lwip_md5_update(&mdContext, (u_char *)secret,
secret_len);
```

Inadequate Encryption Strength\Path 3:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=793
Status New

The application uses a weak cryptographic algorithm, lwip_md5_update at line 1725 of cesanta@@mongoose-newest-CVE-2020-8597-TP.c, to protect sensitive personal information secret_len, from cesanta@@mongoose-newest-CVE-2020-8597-TP.c at line 1725.

	Source	Destination
File	cesanta@@mongoose-newest-CVE-2020-8597-TP.c	cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Line	1877	1877
Object	secret_len	lwip_md5_update

Code Snippet
File Name cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Method static void eap_response(ppp_pcb *pcb, u_char *inp, int id, int len) {

```
....
1877.             lwip_md5_update(&mdContext, (u_char *)secret,
secret_len);
```

Inadequate Encryption Strength\Path 4:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=794
Status New

The application uses a weak cryptographic algorithm, lwip_md5_update at line 1725 of cesanta@@mongoose-newest-CVE-2020-8597-TP.c, to protect sensitive personal information secret, from cesanta@@mongoose-newest-CVE-2020-8597-TP.c at line 1725.

	Source	Destination
File	cesanta@@mongoose-newest-CVE-2020-8597-TP.c	cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Line	1877	1877
Object	secret	lwip_md5_update

Code Snippet
File Name cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Method static void eap_response(ppp_pcb *pcb, u_char *inp, int id, int len) {

```
....
1877.             lwip_md5_update(&mdContext, (u_char *)secret,
secret_len);
```

Inadequate Encryption Strength\Path 5:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=795
Status New

The application uses a weak cryptographic algorithm, SHA1Update at line 315 of cesanta@@mongoose-newest-CVE-2020-8597-TP.c, to protect sensitive personal information pn_secret, from cesanta@@mongoose-newest-CVE-2020-8597-TP.c at line 315.

	Source	Destination
File	cesanta@@mongoose-newest-CVE-2020-8597-TP.c	cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Line	328	328
Object	pn_secret	SHA1Update

Code Snippet
File Name cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Method pncrypt_setkey(int timeoffs)

```
....
328.        SHA1Update(&ctxt, pn_secret, strlen(pn_secret));
```

Inadequate Encryption Strength\Path 6:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=796
Status New

The application uses a weak cryptographic algorithm, SHA1Update at line 315 of cesanta@@mongoose-newest-CVE-2020-8597-TP.c, to protect sensitive personal information pn_secret, from cesanta@@mongoose-newest-CVE-2020-8597-TP.c at line 315.

	Source	Destination
File	cesanta@@mongoose-newest-CVE-2020-8597-TP.c	cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Line	328	328
Object	pn_secret	SHA1Update

Code Snippet
File Name cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Method pncrypt_setkey(int timeoffs)

```
....
328.        SHA1Update(&ctxt, pn_secret, strlen(pn_secret));
```

Use After Free

Query Path:
CPP\Cx\CPP Medium Threat\Use After Free Version:1

Categories

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

Description

Use After Free\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=852
Status	New

The pointer prog at ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c in line 936 is being used after it has been freed.

	Source	Destination
File	ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c	ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c
Line	947	940
Object	p	prog

Code Snippet
File Name ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c
Method static void *default_alloc(void *ctx, void *p, int n)

```
....
947.                free(p);
```

File Name	ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c
Method	void regfreex(void *(*alloc)(void *ctx, void *p, int n), void *ctx, Reprog *prog)

```
....
940.              alloc(ctx, prog, 0);
```

Use After Free\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=853
Status	New

The pointer key at chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c in line 86 is being used after it has been freed.

	Source	Destination
File	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Line	115	114
Object	key	key

Code Snippet

File Name	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Method	static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
115.      free(key);
....
114.      setenv(key, argv[i], 1);
```

Use After Free\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=854
Status	New

The pointer key at chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c in line 86 is being used after it has been freed.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Line	115	114

Object	key	key

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
115.      free(key);
....
114.      setenv(key, argv[i], 1);
```

Use After Free\Path 4:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=855
Status New

The pointer key at chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c in line 86 is being used after it has been freed.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Line	115	114
Object	key	key

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
115.      free(key);
....
114.      setenv(key, argv[i], 1);
```

Use After Free\Path 5:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=856
Status New

The pointer key at chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c in line 86 is being used after it has been freed.

	Source	Destination
File	chromium@@chromium-117.0.5881.1-	chromium@@chromium-117.0.5881.1-

	CVE-2021-44109-FP.c	CVE-2021-44109-FP.c
Line	115	114
Object	key	key

Code Snippet
File Name chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
115.      free(key);
....
114.      setenv(key, argv[i], 1);
```

Use After Free\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=857
Status	New

The pointer key at chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c in line 86 is being used after it has been freed.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Line	115	114
Object	key	key

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
115.      free(key);
....
114.      setenv(key, argv[i], 1);
```

Environment Injection

Query Path:
CPP\Cx\CPP Medium Threat\Environment Injection Version:0

Categories

OWASP Top 10 2013: A1-Injection
FISMA 2014: System And Information Integrity
NIST SP 800-53: SI-10 Information Input Validation (P1)
OWASP Top 10 2017: A1-Injection

Description

Environment Injection\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=731
Status	New

	Source	Destination
File	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Line	121	121
Object	getenv	setenv

Code Snippet
File Name chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
121.    setenv("ARG0", getenv("SRC"), 0);
```

Environment Injection\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=732
Status	New

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Line	121	121
Object	getenv	setenv

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
121.    setenv("ARG0", getenv("SRC"), 0);
```

Environment Injection\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=733
Status	New

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Line	121	121
Object	getenv	setenv

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
121.    setenv("ARG0", getenv("SRC"), 0);
```

Environment Injection\Path 4:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=734
Status New

	Source	Destination
File	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Line	121	121
Object	getenv	setenv

Code Snippet
File Name chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
121.    setenv("ARG0", getenv("SRC"), 0);
```

Environment Injection\Path 5:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=735
Status New

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Line	121	121

Object	getenv	setenv

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Method static PP_Bool Instance_DidCreate(PP_Instance instance,

```
....
121.    setenv("ARG0", getenv("SRC"), 0);
```

Double Free

Query Path:
CPP\Cx\CPP Medium Threat\Double Free Version:1

Categories

NIST SP 800-53: SI-16 Memory Protection (P1)

Description
Double Free\Path 1:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=730
Status New

	Source	Destination
File	ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c	ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c
Line	947	950
Object	p	p

Code Snippet
File Name ccxvii@@mujs-1.2.0-CVE-2022-30974-TP.c
Method static void *default_alloc(void *ctx, void *p, int n)

```
....
947.                free(p);
....
950.        return realloc(p, (size_t)n);
```

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=1000009&projectid=5&pathid=2135
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Line	353	353
Object	l	l

Code Snippet
File Name bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Method check_syslog_udp (void (*syslog_send)(int), int options,

```
....
353.        buf[l] = '\0';
```

Unchecked Array Index\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2136
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Line	353	353
Object	l	l

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Method check_syslog_udp (void (*syslog_send)(int), int options,

```
....
353.        buf[l] = '\0';
```

Unchecked Array Index\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2137
Status	New

	Source	Destination

File	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Line	353	353
Object	l	l

Code Snippet
File Name bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Method check_syslog_udp (void (*syslog_send)(int), int options,

```
....
353.        buf[l] = '\0';
```

Unchecked Array Index\Path 4:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2138
Status New

	Source	Destination
File	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Line	353	353
Object	l	l

Code Snippet
File Name bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Method check_syslog_udp (void (*syslog_send)(int), int options,

```
....
353.        buf[l] = '\0';
```

Unchecked Array Index\Path 5:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2139
Status New

	Source	Destination
File	ccxvii@@mujs-1.0.7-CVE-2021-45005-FP.c	ccxvii@@mujs-1.0.7-CVE-2021-45005-FP.c
Line	266	266
Object	inst	inst

Code Snippet
File Name ccxvii@@mujs-1.0.7-CVE-2021-45005-FP.c
Method static void labelto(JF, int inst, int addr)

```
....
266.        F->code[inst] = addr;
```

Unchecked Array Index\Path 6:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2140
Status New

	Source	Destination
File	ccxvii@@mujs-1.0.8-CVE-2021-45005-FP.c	ccxvii@@mujs-1.0.8-CVE-2021-45005-FP.c
Line	266	266
Object	inst	inst

Code Snippet
File Name ccxvii@@mujs-1.0.8-CVE-2021-45005-FP.c
Method static void labelto(JF, int inst, int addr)

```
....
266.        F->code[inst] = addr;
```

Unchecked Array Index\Path 7:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2141
Status New

	Source	Destination
File	ccxvii@@mujs-1.1.0-CVE-2021-45005-FP.c	ccxvii@@mujs-1.1.0-CVE-2021-45005-FP.c
Line	266	266
Object	inst	inst

Code Snippet
File Name ccxvii@@mujs-1.1.0-CVE-2021-45005-FP.c
Method static void labelto(JF, int inst, int addr)

```
....
266.        F->code[inst] = addr;
```

Unchecked Array Index\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2142
Status	New

	Source	Destination
File	ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c	ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c
Line	250	250
Object	inst	inst

Code Snippet
File Name ccxvii@@mujs-1.1.3-CVE-2021-45005-TP.c
Method static void labelto(JF, int inst, int addr)

```
....
250.        F->code[inst] = addr;
```

Unchecked Array Index\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2143
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	3309	3309
Object	k	k

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void dstsub(int n, double *a, int nc, double *c)

```
....
3309.          a[k] = wkr * a[k] + wki * a[j];
```

Unchecked Array Index\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2144

Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	496	496
Object	k	k

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void dfct(int n, double *a, double *t, int *ip, double *w)

```
....
496.                a[k] = yr;
```

Unchecked Array Index\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2145
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	498	498
Object	k	k

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void dfct(int n, double *a, double *t, int *ip, double *w)

```
....
498.                t[k] = xi + yi;
```

Unchecked Array Index\Path 12:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2146
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c

Line	526	526
Object	l	l

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void dfct(int n, double *a, double *t, int *ip, double *w)

```
....
526.              a[l] = t[0] + t[1];
```

Unchecked Array Index\Path 13:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2147
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	538	538
Object	k	k

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void dfct(int n, double *a, double *t, int *ip, double *w)

```
....
538.                  t[k] = t[m + k] + t[m + j];
```

Unchecked Array Index\Path 14:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2148
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	543	543
Object	l	l

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c

Method	void dfct(int n, double *a, double *t, int *ip, double *w)

```
....
543.          a[l] = t[0];
```

Unchecked Array Index\Path 15:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2149
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	584	584
Object	k	k

Code Snippet	
File Name	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method	void dfst(int n, double *a, double *t, int *ip, double *w)

```
....
584.              a[k] = yr;
```

Unchecked Array Index\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2150
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	586	586
Object	k	k

Code Snippet	
File Name	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method	void dfst(int n, double *a, double *t, int *ip, double *w)

```
....
586.              t[k] = xi - yi;
```

Unchecked Array Index\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2151
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	615	615
Object	l	l

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void dfst(int n, double *a, double *t, int *ip, double *w)

```
....
615.              a[l] = t[0] + t[1];
```

Unchecked Array Index\Path 18:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2152
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	627	627
Object	k	k

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void dfst(int n, double *a, double *t, int *ip, double *w)

```
....
627.                  t[k] = t[m + k] - t[m + j];
```

Unchecked Array Index\Path 19:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2153
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	632	632
Object	l	l

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void dfst(int n, double *a, double *t, int *ip, double *w)

```
....
632.            a[l] = t[0];
```

Unchecked Array Index\Path 20:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2154
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	675	675
Object	nw1	nw1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void makewt(int nw, int *ip, double *w)

```
....
675.                w[nw1] = 1;
```

Unchecked Array Index\Path 21:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2155
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	916	916

Object	j1	j1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
916.                    a[j1] = yr;
```

Unchecked Array Index\Path 22:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2156
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	918	918
Object	k1	k1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
918.                    a[k1] = xr;
```

Unchecked Array Index\Path 23:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2157
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	926	926
Object	j1	j1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
926.                    a[j1] = yr;
```

Unchecked Array Index\Path 24:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2158
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	928	928
Object	k1	k1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
928.                    a[k1] = xr;
```

Unchecked Array Index\Path 25:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2159
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	936	936
Object	j1	j1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
936.                    a[j1] = yr;
```

Unchecked Array Index\Path 26:

Severity	Low

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2160
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	938	938
Object	k1	k1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
938.                    a[k1] = xr;
```

Unchecked Array Index\Path 27:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2161
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	946	946
Object	j1	j1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
946.                    a[j1] = yr;
```

Unchecked Array Index\Path 28:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2162
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	948	948
Object	k1	k1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
948.                    a[k1] = xr;
```

Unchecked Array Index\Path 29:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2163
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	956	956
Object	j1	j1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
956.                    a[j1] = yr;
```

Unchecked Array Index\Path 30:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2164
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	958	958

Object	k1	k1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
958.                    a[k1] = xr;
```

Unchecked Array Index\Path 31:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2165
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	966	966
Object	j1	j1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
966.                    a[j1] = yr;
```

Unchecked Array Index\Path 32:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2166
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	968	968
Object	k1	k1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
968.                    a[k1] = xr;
```

Unchecked Array Index\Path 33:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2167
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	976	976
Object	j1	j1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
976.                    a[j1] = yr;
```

Unchecked Array Index\Path 34:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2168
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	978	978
Object	k1	k1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
978.                    a[k1] = xr;
```

Unchecked Array Index\Path 35:

Severity	Low

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2169
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	986	986
Object	j1	j1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
986.                    a[j1] = yr;
```

Unchecked Array Index\Path 36:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2170
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	988	988
Object	k1	k1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
988.                    a[k1] = xr;
```

Unchecked Array Index\Path 37:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2171
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	996	996
Object	j1	j1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
996.                    a[j1] = yr;
```

Unchecked Array Index\Path 38:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2172
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	998	998
Object	k1	k1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
998.                    a[k1] = xr;
```

Unchecked Array Index\Path 39:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2173
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	1006	1006

Object	j1	j1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
1006.                    a[j1] = yr;
```

Unchecked Array Index\Path 40:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2174
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	1008	1008
Object	k1	k1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
1008.                    a[k1] = xr;
```

Unchecked Array Index\Path 41:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2175
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	1016	1016
Object	j1	j1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
1016.                   a[j1] = yr;
```

Unchecked Array Index\Path 42:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2176
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	1018	1018
Object	k1	k1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
1018.                   a[k1] = xr;
```

Unchecked Array Index\Path 43:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2177
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	1026	1026
Object	j1	j1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
1026.                   a[j1] = yr;
```

Unchecked Array Index\Path 44:

Severity	Low

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2178
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	1028	1028
Object	k1	k1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
1028.                    a[k1] = xr;
```

Unchecked Array Index\Path 45:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2179
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	1036	1036
Object	j1	j1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
1036.                    a[j1] = yr;
```

Unchecked Array Index\Path 46:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2180
Status	New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	1038	1038
Object	k1	k1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
1038.                    a[k1] = xr;
```

Unchecked Array Index\Path 47:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2181
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	1046	1046
Object	j1	j1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
1046.                    a[j1] = yr;
```

Unchecked Array Index\Path 48:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2182
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	1048	1048

Object	k1	k1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
1048.                    a[k1] = xr;
```

Unchecked Array Index\Path 49:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2183
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	1056	1056
Object	j1	j1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
1056.                    a[j1] = yr;
```

Unchecked Array Index\Path 50:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2184
Status New

	Source	Destination
File	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c	chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Line	1058	1058
Object	k1	k1

Code Snippet
File Name chromium@@chromium-102.0.4995.0-CVE-2021-3520-FP.c
Method void bitrv2(int n, int *ip, double *a)

```
....
1058.                   a[k1] = xr;
```

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=1000009&projectid=5&pathid=1801
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	836	836
Object	null	function

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c

Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
836.      function->import_func_linked = is_built_in_module ? NULL :
linked_func;
```

NULL Pointer Dereference\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1802
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	779	836
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
779.      WASMFunction *linked_func = NULL;
....
836.      function->import_func_linked = is_built_in_module ? NULL :
linked_func;
```

NULL Pointer Dereference\Path 3:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1803
Status New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	829	829
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
829.      function->func_ptr_linked = is_built_in_module ? linked_func :
NULL;
```

NULL Pointer Dereference\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1804
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	779	829
Object	null	function

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c

Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
779.      WASMFunction *linked_func = NULL;
....
829.      function->func_ptr_linked = is_built_in_module ? linked_func :
NULL;
```

NULL Pointer Dereference\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1805
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-

	52284-FP.c	52284-FP.c
Line	834	834
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
834.      function->import_module = is_built_in_module ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 6:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1806
Status New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 1404 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c in line 770.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	1519	834
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method load_import_section(const uint8 *buf, const uint8 *buf_end, WASMModule *module,

```
....
1519.              WASMModule *sub_module = NULL;
```

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
834.      function->import_module = is_built_in_module ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1807
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1045 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1045.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	1109	1109
Object	null	function

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c

Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1109.      function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1808
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1045 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1045.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	1056	1109

Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1056.       WASMModule *sub_module = NULL;
....
1109.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1809
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1045 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1045.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	1104	1104
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1104.       function->func_ptr_linked = is_native_symbol ? linked_func :
NULL;
```

NULL Pointer Dereference\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1810
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1045 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c in line 1045.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	1110	1110
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1110.       function->import_func_linked = is_native_symbol ? NULL :
linked_func;
```

NULL Pointer Dereference\Path 11:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1811
Status New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1045 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1045.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	1109	1109
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1109.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 12:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1812
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1045 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1045.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	1056	1109
Object	null	function

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c

Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1056.       WASMModule *sub_module = NULL;
....
1109.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 13:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1813
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1045 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1045.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	1104	1104
Object	null	function

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c

Method	load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1104.      function->func_ptr_linked = is_native_symbol ? linked_func :
NULL;
```

NULL Pointer Dereference\Path 14:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1814
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1045 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c in line 1045.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	1110	1110
Object	null	function

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method	load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1110.      function->import_func_linked = is_native_symbol ? NULL :
linked_func;
```

NULL Pointer Dereference\Path 15:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1815
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c in line 709 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c in line 709.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c

Line	782	782
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
782.      function->func_ptr_linked = is_built_in_module ? linked_func :
NULL;
```

NULL Pointer Dereference\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1816
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c in line 709 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c in line 709.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	787	787
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
787.      function->import_module = is_built_in_module ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1817
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c in line 1331 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c in line 709.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	1446	787
Object	null	function

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c

Method load_import_section(const uint8 *buf, const uint8 *buf_end, WASMModule *module,

```
....
1446.              WASMModule *sub_module = NULL;
```

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c

Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
787.      function->import_module = is_built_in_module ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 18:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1818
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c in line 709 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c in line 709.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	789	789
Object	null	function

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Method	load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
789.      function->import_func_linked = is_built_in_module ? NULL :
linked_func;
```

NULL Pointer Dereference\Path 19:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1819
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c in line 1132 is not initialized when it is used by module_inst at bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c in line 1132.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c
Line	1277	1276
Object	null	module_inst

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c
Method	wasm_instantiate(WASMModule *module, bool is_sub_inst,

```
....
1277.        module_inst->memory_count ? module_inst->memories[0] :
NULL;
....
1276.      module_inst->default_memory =
```

NULL Pointer Dereference\Path 20:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1820
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c in line 1132 is not initialized when it is used by module_inst at bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c in line 1132.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c
Line	1362	1361
Object	null	module_inst

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c
Method wasm_instantiate(WASMModule *module, bool is_sub_inst,

```
....
1362.         module_inst->table_count ? module_inst->tables[0] : NULL;
....
1361.       module_inst->default_table =
```

NULL Pointer Dereference\Path 21:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1821
Status New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c in line 1043 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c in line 1043.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	1116	1116
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1116.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 22:

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

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1822
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c in line 1043 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c in line 1043.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	1055	1116
Object	null	function

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c

Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1055.       WASMModule *sub_module = NULL;
....
1116.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 23:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1823
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c in line 1043 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c in line 1043.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	1111	1111
Object	null	function

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c

Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1111.       function->func_ptr_linked = is_native_symbol ? linked_func :
NULL;
```

NULL Pointer Dereference\Path 24:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1824
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c in line 1043 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c in line 1043.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	1117	1117
Object	null	function

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c

Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1117.       function->import_func_linked = is_native_symbol ? NULL :
linked_func;
```

NULL Pointer Dereference\Path 25:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1825
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c in line 773 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c in line 773.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Line	840	840

Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
840.      function->func_ptr_linked = is_built_in_module ? linked_func :
NULL;
```

NULL Pointer Dereference\Path 26:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1826
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c in line 773 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c in line 773.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Line	845	845
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
845.      function->import_module = is_built_in_module ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 27:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1827
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c in line 1433 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c in line 773.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Line	1548	845
Object	null	function

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c

Method load_import_section(const uint8 *buf, const uint8 *buf_end, WASMModule *module,

```
....
1548.              WASMModule *sub_module = NULL;
```

File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c

Method load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
845.      function->import_module = is_built_in_module ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 28:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1828
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c in line 773 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c in line 773.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Line	847	847
Object	null	function

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Method	load_function_import(const WASMModule *parent_module, WASMModule *sub_module,

```
....
847.      function->import_func_linked = is_built_in_module ? NULL :
linked_func;
```

NULL Pointer Dereference\Path 29:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1829
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c in line 1078 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c in line 1078.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Line	1142	1142
Object	null	function

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Method	load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1142.      function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 30:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1830
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c in line 1078 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c in line 1078.

	Source	Destination

File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Line	1089	1142
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1089.       WASMModule *sub_module = NULL;
....
1142.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 31:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1831
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c in line 1078 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c in line 1078.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Line	1137	1137
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1137.       function->func_ptr_linked = is_native_symbol ? linked_func :
NULL;
```

NULL Pointer Dereference\Path 32:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1832
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c in line 1078 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c in line 1078.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Line	1143	1143
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1143.       function->import_func_linked = is_native_symbol ? NULL :
linked_func;
```

NULL Pointer Dereference\Path 33:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1833
Status New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c in line 1078 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c in line 1078.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Line	1142	1142
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1142.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 34:

Severity Low

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1834
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c in line 1078 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c in line 1078.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Line	1089	1142
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1089.       WASMModule *sub_module = NULL;
....
1142.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 35:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1835
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c in line 1078 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c in line 1078.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Line	1137	1137
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1137.       function->func_ptr_linked = is_native_symbol ? linked_func :
NULL;
```

NULL Pointer Dereference\Path 36:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1836
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c in line 1078 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c in line 1078.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Line	1143	1143
Object	null	function

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c

Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1143.       function->import_func_linked = is_native_symbol ? NULL :
linked_func;
```

NULL Pointer Dereference\Path 37:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1837
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c in line 1054 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c in line 1054.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Line	1118	1118

Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1118.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 38:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1838
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c in line 1054 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c in line 1054.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Line	1065	1118
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1065.       WASMModule *sub_module = NULL;
....
1118.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 39:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1839
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c in line 1054 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c in line 1054.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Line	1113	1113
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1113.       function->func_ptr_linked = is_native_symbol ? linked_func :
NULL;
```

NULL Pointer Dereference\Path 40:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1840
Status New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c in line 1054 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c in line 1054.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Line	1119	1119
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1119.       function->import_func_linked = is_native_symbol ? NULL :
linked_func;
```

NULL Pointer Dereference\Path 41:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1841
Status New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c in line 1054 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c in line 1054.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Line	1118	1118
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1118.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 42:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1842
Status New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c in line 1054 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c in line 1054.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Line	1065	1118
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1065.       WASMModule *sub_module = NULL;
....
1118.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 43:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1843
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c in line 1054 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c in line 1054.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Line	1113	1113
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1113.       function->func_ptr_linked = is_native_symbol ? linked_func :
NULL;
```

NULL Pointer Dereference\Path 44:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1844
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c in line 1054 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c in line 1054.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Line	1119	1119
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1119.       function->import_func_linked = is_native_symbol ? NULL :
linked_func;
```

NULL Pointer Dereference\Path 45:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1845
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c in line 1064 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c in line 1064.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Line	1128	1128
Object	null	function

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c

Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1128.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 46:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1846
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c in line 1064 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c in line 1064.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Line	1075	1128

Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1075.       WASMModule *sub_module = NULL;
....
1128.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 47:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1847
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c in line 1064 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c in line 1064.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Line	1123	1123
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1123.       function->func_ptr_linked = is_native_symbol ? linked_func :
NULL;
```

NULL Pointer Dereference\Path 48:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1848
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c in line 1064 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c in line 1064.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Line	1129	1129
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1129.       function->import_func_linked = is_native_symbol ? NULL :
linked_func;
```

NULL Pointer Dereference\Path 49:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1849
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c in line 1064 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c in line 1064.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c
Line	1128	1128
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1128.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

NULL Pointer Dereference\Path 50:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1850
Status	New

The variable declared in null at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c in line 1064 is not initialized when it is used by function at bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c in line 1064.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c
Line	1075	1128
Object	null	function

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c
Method load_function_import(const uint8 **p_buf, const uint8 *buf_end,

```
....
1075.       WASMModule *sub_module = NULL;
....
1128.       function->import_module = is_native_symbol ? NULL :
sub_module;
```

Unchecked Return Value

Query Path:
CPP\Cx\CPP Low Visibility\Unchecked Return Value Version:1

Categories

NIST SP 800-53: SI-11 Error Handling (P2)

Description
Unchecked Return Value\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1403
Status	New

The check_openlog_message method calls the snprintf function, at line 223 of bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Line	233	233
Object	snprintf	snprintf

Code Snippet
File Name bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c

Method	check_openlog_message (const struct msg_t *msg, int msgnum,

```
....
233.    snprintf (expected_ident, sizeof (expected_ident),
"%s%s%.0d%s:",
```

Unchecked Return Value\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1404
Status	New

The check_openlog_message_large method calls the snprintf function, at line 255 of bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Line	259	259
Object	snprintf	snprintf

Code Snippet	
File Name	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Method	check_openlog_message_large (const struct msg_t *msg, int msgnum,

```
....
259.    snprintf (expected_ident, sizeof (expected_ident),
"%s%s%.0d%s:",
```

Unchecked Return Value\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1405
Status	New

The check_syslog_console_read_large method calls the fgets function, at line 462 of bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Line	465	465
Object	fgets	fgets

Code Snippet	
File Name	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Method	check_syslog_console_read_large (FILE *fp)

```
....
465.    TEST_VERIFY (fgets (buf, sizeof (buf), fp) != NULL);
```

Unchecked Return Value\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1406
Status	New

The check_openlog_message method calls the snprintf function, at line 223 of bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Line	233	233
Object	snprintf	snprintf

Code Snippet	
File Name	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Method	check_openlog_message (const struct msg_t *msg, int msgnum,

```
....
233.    snprintf (expected_ident, sizeof (expected_ident),
"%s%s%.0d%s:",
```

Unchecked Return Value\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1407
Status	New

The check_openlog_message_large method calls the snprintf function, at line 255 of bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Line	259	259

Object	snprintf	snprintf

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Method check_openlog_message_large (const struct msg_t *msg, int msgnum,

```
....
259.    snprintf (expected_ident, sizeof (expected_ident),
"%s%s%.0d%s:",
```

Unchecked Return Value\Path 6:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1408
Status New

The check_syslog_console_read_large method calls the fgets function, at line 462 of bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Line	465	465
Object	fgets	fgets

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read_large (FILE *fp)

```
....
465.    TEST_VERIFY (fgets (buf, sizeof (buf), fp) != NULL);
```

Unchecked Return Value\Path 7:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1409
Status New

The check_openlog_message method calls the snprintf function, at line 223 of bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c

Line	233	233
Object	snprintf	snprintf

Code Snippet
File Name bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Method check_openlog_message (const struct msg_t *msg, int msgnum,

```
....
233.    snprintf (expected_ident, sizeof (expected_ident),
"%s%s%.0d%s:",
```

Unchecked Return Value\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1410
Status	New

The check_openlog_message_large method calls the snprintf function, at line 255 of bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Line	259	259
Object	snprintf	snprintf

Code Snippet
File Name bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Method check_openlog_message_large (const struct msg_t *msg, int msgnum,

```
....
259.    snprintf (expected_ident, sizeof (expected_ident),
"%s%s%.0d%s:",
```

Unchecked Return Value\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1411
Status	New

The check_syslog_console_read_large method calls the fgets function, at line 462 of bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination

File	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Line	465	465
Object	fgets	fgets

Code Snippet
File Name bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read_large (FILE *fp)

```
....
465.    TEST_VERIFY (fgets (buf, sizeof (buf), fp) != NULL);
```

Unchecked Return Value\Path 10:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1412
Status New

The check_openlog_message method calls the snprintf function, at line 223 of bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Line	233	233
Object	snprintf	snprintf

Code Snippet
File Name bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Method check_openlog_message (const struct msg_t *msg, int msgnum,

```
....
233.    snprintf (expected_ident, sizeof (expected_ident),
"%s%s%.0d%s:",
```

Unchecked Return Value\Path 11:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1413
Status New

The check_openlog_message_large method calls the snprintf function, at line 255 of bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Line	259	259
Object	snprintf	snprintf

Code Snippet
File Name bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Method check_openlog_message_large (const struct msg_t *msg, int msgnum,

```
....
259.    snprintf (expected_ident, sizeof (expected_ident),
"%s%s%.0d%s:",
```

Unchecked Return Value\Path 12:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1414
Status New

The check_syslog_console_read_large method calls the fgets function, at line 462 of bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Line	465	465
Object	fgets	fgets

Code Snippet
File Name bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read_large (FILE *fp)

```
....
465.    TEST_VERIFY (fgets (buf, sizeof (buf), fp) != NULL);
```

Unchecked Return Value\Path 13:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1415
Status New

The module_reader_callback method calls the snprintf function, at line 186 of bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Line	197	197
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Method module_reader_callback(const char *module_name, uint8 **p_buffer,

```
....
197.      snprintf(wasm_file_name, sz, format, module_search_path,
module_name);
```

Unchecked Return Value\Path 14:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1416
Status New

The wasm_loader_prepare_bytecode method calls the snprintf function, at line 5682 of bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	7537	7537
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
7537.                                 snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 15:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1417
Status	New

The set_error_buf method calls the snprintf function, at line 21 of bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	24	24
Object	snprintf	snprintf

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c

Method set_error_buf(char *error_buf, uint32 error_buf_size, const char *string)

```
....
24.          snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1418
Status	New

The set_error_buf_v method calls the snprintf function, at line 30 of bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	40	40
Object	snprintf	snprintf

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c

Method set_error_buf_v(char *error_buf, uint32 error_buf_size,

```
....
40.          snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1419
Status	New

The wasm_loader_prepare_bytecode method calls the snprintf function, at line 6444 of bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	9076	9076
Object	snprintf	snprintf

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c

Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
9076.                                snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 18:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1420
Status	New

The set_error_buf method calls the snprintf function, at line 25 of bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	28	28

Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method set_error_buf(char *error_buf, uint32 error_buf_size, const char *string)

```
....
28.          snprintf(error_buf, error_buf_size, "WASM module load
failed: %s",
```

Unchecked Return Value\Path 19:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1421
Status New

The set_error_buf_v method calls the snprintf function, at line 34 of bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	43	43
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method set_error_buf_v(char *error_buf, uint32 error_buf_size, const char *format, ...)

```
....
43.          snprintf(error_buf, error_buf_size, "WASM module load
failed: %s", buf);
```

Unchecked Return Value\Path 20:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1422
Status New

The wasm_loader_prepare_bytecode method calls the snprintf function, at line 6444 of bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	9076	9076
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
9076.                                   snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 21:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1423
Status New

The set_error_buf method calls the snprintf function, at line 25 of bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	28	28
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method set_error_buf(char *error_buf, uint32 error_buf_size, const char *string)

```
....
28.          snprintf(error_buf, error_buf_size, "WASM module load
failed: %s",
```

Unchecked Return Value\Path 22:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1424

Status	New

The set_error_buf_v method calls the snprintf function, at line 34 of bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	43	43
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method set_error_buf_v(char *error_buf, uint32 error_buf_size, const char *format, ...)

```
....
43.           snprintf(error_buf, error_buf_size, "WASM module load
failed: %s", buf);
```

Unchecked Return Value\Path 23:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1425
Status	New

The wasm_loader_prepare_bytecode method calls the snprintf function, at line 4787 of bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	5943	5943
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
5943.                          snprintf(msg, 128, "WASM loader prepare
bytecode failed: "
```

Unchecked Return Value\Path 24:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1426
Status	New

The wasm_loader_prepare_bytecode method calls the snprintf function, at line 4787 of bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	6017	6017
Object	snprintf	snprintf

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c

Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
6017.                          snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 25:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1427
Status	New

The wasm_loader_prepare_bytecode method calls the snprintf function, at line 4787 of bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c

Line	6027	6027
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
6027.                       snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 26:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1428
Status	New

The set_error_buf method calls the snprintf function, at line 21 of bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	24	24
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Method set_error_buf(char *error_buf, uint32 error_buf_size, const char *string)

```
....
24.          snprintf(error_buf, error_buf_size, "%s", string);
```

Unchecked Return Value\Path 27:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1429
Status	New

The wasm_loader_find_block_addr method calls the snprintf function, at line 2910 of bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	3258	3258
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Method wasm_loader_find_block_addr(BlockAddr *block_addr_cache,

```
....
3258.                               snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 28:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1430
Status New

The wasm_loader_find_block_addr method calls the snprintf function, at line 2910 of bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	3268	3268
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Method wasm_loader_find_block_addr(BlockAddr *block_addr_cache,

```
....
3268.                           snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 29:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1431

Status	New

The check_stack_top_values method calls the snprintf function, at line 3489 of bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	3513	3513
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Method check_stack_top_values(uint8 *frame_ref, int32 stack_cell_num, uint8 type,

```
....
3513.              snprintf(error_buf, error_buf_size, "%s%s%s",
```

Unchecked Return Value\Path 30:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1432
Status	New

The set_error_buf method calls the snprintf function, at line 21 of bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c
Line	24	24
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c
Method set_error_buf(char *error_buf, uint32 error_buf_size, const char *string)

```
....
24.          snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 31:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1433
Status	New

The set_error_buf_v method calls the snprintf function, at line 30 of bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c
Line	40	40
Object	snprintf	snprintf

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c

Method set_error_buf_v(char *error_buf, uint32 error_buf_size,

```
....
40.          snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 32:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1434
Status	New

The wasm_set_exception method calls the snprintf function, at line 1748 of bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c
Line	1752	1752
Object	snprintf	snprintf

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c

Method wasm_set_exception(WASMModuleInstance *module_inst,

```
....
1752.          snprintf(module_inst->cur_exception,
```

Unchecked Return Value\Path 33:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1435
Status	New

The set_error_buf method calls the snprintf function, at line 21 of bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	24	24
Object	snprintf	snprintf

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c

Method set_error_buf(char *error_buf, uint32 error_buf_size, const char *string)

```
....
24.          snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 34:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1436
Status	New

The set_error_buf_v method calls the snprintf function, at line 30 of bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	40	40
Object	snprintf	snprintf

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Method	set_error_buf_v(char *error_buf, uint32 error_buf_size,

```
....
40.          snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 35:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1437
Status	New

The wasm_loader_prepare_bytecode method calls the snprintf function, at line 6137 of bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	8403	8403
Object	snprintf	snprintf

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Method	wasm_loader_prepare_bytecode(WASMModule *module,

```
....
8403.                                snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 36:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1438
Status	New

The module_reader_callback method calls the snprintf function, at line 186 of bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination

File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c
Line	197	197
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c
Method module_reader_callback(const char *module_name, uint8 **p_buffer,

```
....
197.      snprintf(wasm_file_name, sz, format, module_search_path,
module_name);
```

Unchecked Return Value\Path 37:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1439
Status New

The set_error_buf method calls the snprintf function, at line 21 of bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Line	24	24
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Method set_error_buf(char *error_buf, uint32 error_buf_size, const char *string)

```
....
24.          snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 38:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1440
Status New

The set_error_buf_v method calls the snprintf function, at line 30 of bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Line	40	40
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Method set_error_buf_v(char *error_buf, uint32 error_buf_size,

```
....
40.          snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 39:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1441
Status New

The wasm_loader_prepare_bytecode method calls the snprintf function, at line 6625 of bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Line	9268	9268
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
9268.                               snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 40:

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

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1442
Status	New

The set_error_buf method calls the snprintf function, at line 29 of bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Line	32	32
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Method set_error_buf(char *error_buf, uint32 error_buf_size, const char *string)

```
....
32.          snprintf(error_buf, error_buf_size, "WASM module load
failed: %s",
```

Unchecked Return Value\Path 41:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1443
Status	New

The set_error_buf_v method calls the snprintf function, at line 38 of bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Line	47	47
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Method set_error_buf_v(char *error_buf, uint32 error_buf_size, const char *format, ...)

```
....
47.          snprintf(error_buf, error_buf_size, "WASM module load
failed: %s", buf);
```

Unchecked Return Value\Path 42:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1444
Status	New

The wasm_loader_prepare_bytecode method calls the snprintf function, at line 6625 of bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Line	9268	9268
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
9268.                                   snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 43:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1445
Status	New

The set_error_buf method calls the snprintf function, at line 29 of bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Line	32	32
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Method set_error_buf(char *error_buf, uint32 error_buf_size, const char *string)

```
....
32.          snprintf(error_buf, error_buf_size, "WASM module load
failed: %s",
```

Unchecked Return Value\Path 44:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1446
Status	New

The set_error_buf_v method calls the snprintf function, at line 38 of bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Line	47	47
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Method set_error_buf_v(char *error_buf, uint32 error_buf_size, const char *format, ...)

```
....
47.          snprintf(error_buf, error_buf_size, "WASM module load
failed: %s", buf);
```

Unchecked Return Value\Path 45:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1447
Status	New

The wasm_loader_prepare_bytecode method calls the snprintf function, at line 6845 of bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Line	9535	9535

Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
9535.                                   snprintf(error_buf, error_buf_size,
```

Unchecked Return Value\Path 46:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1448
Status	New

The set_error_buf method calls the snprintf function, at line 32 of bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Line	35	35
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Method set_error_buf(char *error_buf, uint32 error_buf_size, const char *string)

```
....
35.          snprintf(error_buf, error_buf_size, "WASM module load
failed: %s",
```

Unchecked Return Value\Path 47:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1449
Status	New

The set_error_buf_v method calls the snprintf function, at line 41 of bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-	bytecodealliance@@wasm-micro-

	runtime-WAMR-1.1.2-CVE-2023-48105-FP.c	runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Line	50	50
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Method set_error_buf_v(char *error_buf, uint32 error_buf_size, const char *format, ...)

```
....
50.          snprintf(error_buf, error_buf_size, "WASM module load
failed: %s", buf);
```

Unchecked Return Value\Path 48:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1450
Status	New

The orcjit_thread_callback method calls the snprintf function, at line 2930 of bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Line	2951	2951
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Method orcjit_thread_callback(void *arg)

```
....
2951.          snprintf(func_name, sizeof(func_name), "%s%d%s",
AOT_FUNC_PREFIX, i,
```

Unchecked Return Value\Path 49:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1451
Status	New

The compile_llvm_jit_functions method calls the snprintf function, at line 2996 of bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Line	3069	3069
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Method compile_llvm_jit_functions(WASMModule *module, char *error_buf,

```
....
3069.          snprintf(func_name, sizeof(func_name), "%s%d",
AOT_FUNC_PREFIX, i);
```

Unchecked Return Value\Path 50:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1452
Status New

The wasm_loader_prepare_bytecode method calls the snprintf function, at line 6845 of bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Line	9535	9535
Object	snprintf	snprintf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
9535.                              snprintf(error_buf, error_buf_size,
```

Use of Sizeof On a Pointer Type

Query Path:
CPP\Cx\CPP Low Visibility\Use of Sizeof On a Pointer Type Version:1
Description

Use of Sizeof On a Pointer Type\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1618
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Line	92	92
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c

Method split_string(char *str, int *count)

```
....
92.          res = (char **)realloc(res, sizeof(char *) * (uint32)(idx + 1));
```

Use of Sizeof On a Pointer Type\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1619
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Line	274	274
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c

Method main(int argc, char *argv[])

```
....
274.              if (dir_list_size >= sizeof(dir_list) / sizeof(char *)) {
```

Use of Sizeof On a Pointer Type\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1620
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Line	276	276
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
276.                            (int)(sizeof(dir_list) / sizeof(char *)));
```

Use of Sizeof On a Pointer Type\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1621
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Line	286	286
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
286.                if (env_list_size >= sizeof(env_list) / sizeof(char
*)) {
```

Use of Sizeof On a Pointer Type\Path 5:

Severity	Low
Result State	To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1622
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Line	288	288
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
288.                            (int)(sizeof(env_list) / sizeof(char *)));
```

Use of Sizeof On a Pointer Type\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1623
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	448	448
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method load_type_section(const uint8 *buf, const uint8 *buf_end, WASMModule *module,

```
....
448.          total_size = sizeof(WASMType*) * (uint64)type_count;
```

Use of Sizeof On a Pointer Type\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&

	pathid=1624
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	1719	1719
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method load_function_section(const uint8 *buf, const uint8 *buf_end,

```
....
1719.          total_size = sizeof(WASMFunction*) * (uint64)func_count;
```

Use of Sizeof On a Pointer Type\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1625
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	2208	2208
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method load_data_segment_section(const uint8 *buf, const uint8 *buf_end,

```
....
2208.          total_size = sizeof(WASMDataSeg*) *
(uint64)data_seg_count;
```

Use of Sizeof On a Pointer Type\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1626
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	4384	4384
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method wasm_loader_emit_ptr(WASMLoaderContext *ctx, void *value)

```
....
4384.          ctx->p_code_compiled += sizeof(void *);
```

Use of Sizeof On a Pointer Type\Path 10:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1627
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	4387	4387
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method wasm_loader_emit_ptr(WASMLoaderContext *ctx, void *value)

```
....
4387.          ctx->code_compiled_size += sizeof(void *);
```

Use of Sizeof On a Pointer Type\Path 11:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1628
Status New

	Source	Destination

File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	6373	6373
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
6373.                                    *(void**)(loader_ctx-
>p_code_compiled - 2 - sizeof(void*)) =
```

Use of Sizeof On a Pointer Type\Path 12:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1629
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Line	82	82
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Method split_string(char *str, int *count)

```
....
82.          res = (char**) realloc(res, sizeof(char*) * (uint32)(idx +
1));
```

Use of Sizeof On a Pointer Type\Path 13:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1630
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-	bytecodealliance@@wasm-micro-

	runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c	runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Line	193	193
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Method int main(int argc, char *argv[])

```
....
193.              if (dir_list_size >= sizeof(dir_list) / sizeof(char*))
{
```

Use of Sizeof On a Pointer Type\Path 14:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1631
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Line	195	195
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Method int main(int argc, char *argv[])

```
....
195.                              (int)(sizeof(dir_list) /
sizeof(char*)));
```

Use of Sizeof On a Pointer Type\Path 15:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1632
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-

	48105-FP.c	48105-FP.c
Line	205	205
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Method int main(int argc, char *argv[])

```
....
205.              if (env_list_size >= sizeof(env_list) / sizeof(char*))
{
```

Use of Sizeof On a Pointer Type\Path 16:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1633
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Line	207	207
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Method int main(int argc, char *argv[])

```
....
207.                              (int)(sizeof(env_list) /
sizeof(char*)));
```

Use of Sizeof On a Pointer Type\Path 17:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1634
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c

Line	543	543
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method load_type_section(const uint8 *buf, const uint8 *buf_end, WASMModule *module,

```
....
543.          total_size = sizeof(WASMType *) * (uint64)type_count;
```

Use of Sizeof On a Pointer Type\Path 18:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1635
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	1843	1843
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method load_function_section(const uint8 *buf, const uint8 *buf_end,

```
....
1843.          total_size = sizeof(WASMFunction *) * (uint64)func_count;
```

Use of Sizeof On a Pointer Type\Path 19:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1636
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	2533	2533

Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method load_data_segment_section(const uint8 *buf, const uint8 *buf_end,

```
....
2533.           total_size = sizeof(WASMDataSeg *) *
(uint64)data_seg_count;
```

Use of Sizeof On a Pointer Type\Path 20:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1637
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	5091	5091
Object	sizeof	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method wasm_loader_emit_ptr(WASMLoaderContext *ctx, void *value)

```
....
5091.           ctx->p_code_compiled += sizeof(void *);
```

Use of Sizeof On a Pointer Type\Path 21:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1638
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	5097	5097
Object	sizeof	sizeof

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method	wasm_loader_emit_ptr(WASMLoaderContext *ctx, void *value)

```
....
5097.          ctx->code_compiled_size += sizeof(void *);
```

Use of Sizeof On a Pointer Type\Path 22:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1639
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	7227	7227
Object	sizeof	sizeof

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method	wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
7227.                                              - sizeof(void *)) =
```

Use of Sizeof On a Pointer Type\Path 23:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1640
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	7330	7330
Object	sizeof	sizeof

Code Snippet

File Name	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method	wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
7330.                           *(void **)(p_code_compiled_tmp -
sizeof(void *)) =
```

Use of Sizeof On a Pointer Type\Path 24:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1641
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	543	543
Object	sizeof	sizeof

Code Snippet

File Name	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method	load_type_section(const uint8 *buf, const uint8 *buf_end, WASMModule *module,

```
....
543.          total_size = sizeof(WASMType *) * (uint64)type_count;
```

Use of Sizeof On a Pointer Type\Path 25:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1642
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	1843	1843
Object	sizeof	sizeof

Code Snippet

File Name	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method	load_function_section(const uint8 *buf, const uint8 *buf_end,

```
....
1843.          total_size = sizeof(WASMFunction *) * (uint64)func_count;
```

Use of Sizeof On a Pointer Type\Path 26:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1643
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	2533	2533
Object	sizeof	sizeof

Code Snippet

File Name	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method	load_data_segment_section(const uint8 *buf, const uint8 *buf_end,

```
....
2533.          total_size = sizeof(WASMDataSeg *) *
(uint64)data_seg_count;
```

Use of Sizeof On a Pointer Type\Path 27:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1644
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	5091	5091
Object	sizeof	sizeof

Code Snippet

File Name	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c

Method	wasm_loader_emit_ptr(WASMLoaderContext *ctx, void *value)

```
....
5091.          ctx->p_code_compiled += sizeof(void *);
```

Use of Sizeof On a Pointer Type\Path 28:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1645
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	5097	5097
Object	sizeof	sizeof

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method	wasm_loader_emit_ptr(WASMLoaderContext *ctx, void *value)

```
....
5097.          ctx->code_compiled_size += sizeof(void *);
```

Use of Sizeof On a Pointer Type\Path 29:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1646
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	7227	7227
Object	sizeof	sizeof

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method	wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
7227.                                                - sizeof(void *)) =
```

Use of Sizeof On a Pointer Type\Path 30:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1647
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	7330	7330
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c

Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
7330.                            *(void **)(p_code_compiled_tmp -
sizeof(void *)) =
```

Use of Sizeof On a Pointer Type\Path 31:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1648
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c
Line	84	84
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c

Method split_string(char *str, int *count)

```
....
84.          res = (char**) realloc(res, sizeof(char*) * (uint32)(idx +
1));
```

Use of Sizeof On a Pointer Type\Path 32:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1649
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c
Line	207	207
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c

Method int main(int argc, char *argv[])

```
....
207.              if (dir_list_size >= sizeof(dir_list) / sizeof(char*))
{
```

Use of Sizeof On a Pointer Type\Path 33:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1650
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c
Line	209	209
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c

Method int main(int argc, char *argv[])

```
....
209.                            (int)(sizeof(dir_list) / sizeof(char*)));
```

Use of Sizeof On a Pointer Type\Path 34:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1651
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c
Line	219	219
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c

Method int main(int argc, char *argv[])

```
....
219.              if (env_list_size >= sizeof(env_list) / sizeof(char*))
{
```

Use of Sizeof On a Pointer Type\Path 35:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1652
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c
Line	221	221
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c

Method int main(int argc, char *argv[])

```
....
221.                            (int)(sizeof(env_list) / sizeof(char*)));
```

Use of Sizeof On a Pointer Type\Path 36:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1653
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	400	400
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c

Method load_type_section(const uint8 *buf, const uint8 *buf_end, WASMModule *module,

```
....
400.          total_size = sizeof(WASMType*) * (uint64)type_count;
```

Use of Sizeof On a Pointer Type\Path 37:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1654
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	1641	1641
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c

Method load_function_section(const uint8 *buf, const uint8 *buf_end,

```
....
1641.          total_size = sizeof(WASMFunction*) * (uint64)func_count;
```

Use of Sizeof On a Pointer Type\Path 38:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1655
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	2113	2113
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c

Method load_data_segment_section(const uint8 *buf, const uint8 *buf_end,

```
....
2113.          total_size = sizeof(WASMDataSeg*) *
(uint64)data_seg_count;
```

Use of Sizeof On a Pointer Type\Path 39:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1656
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	3939	3939
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c

Method wasm_loader_emit_ptr(WASMLoaderContext *ctx, void *value)

```
....
3939.          ctx->p_code_compiled += sizeof(void *);
```

Use of Sizeof On a Pointer Type\Path 40:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1657
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	3942	3942
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c

Method wasm_loader_emit_ptr(WASMLoaderContext *ctx, void *value)

```
....
3942.          ctx->code_compiled_size += sizeof(void *);
```

Use of Sizeof On a Pointer Type\Path 41:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1658
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	5260	5260
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c

Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
5260.                                    *(void**)(loader_ctx-
>p_code_compiled - 2 - sizeof(void*)) =
```

Use of Sizeof On a Pointer Type\Path 42:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1659
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c
Line	353	353
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c

Method memories_instantiate(const WASMModule *module,

```
....
353.      total_size = sizeof(WASMMemoryInstance*) *
(uint64)memory_count;
```

Use of Sizeof On a Pointer Type\Path 43:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1660
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c
Line	468	468
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c

Method tables_instantiate(const WASMModule *module,

```
....
468.      uint64 total_size = sizeof(WASMTableInstance*) *
(uint64)table_count;
```

Use of Sizeof On a Pointer Type\Path 44:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1661
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c
Line	2311	2311
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c

Method wasm_get_module_mem_consumption(const WASMModule *module,

```
....
2311.      mem_conspn->types_size = sizeof(WASMType *) * module-
>type_count;
```

Use of Sizeof On a Pointer Type\Path 45:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1662
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c
Line	2321	2321
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c

Method wasm_get_module_mem_consumption(const WASMModule *module,

```
....
2321.      mem_conspn->functions_size = sizeof(WASMFunction *)
```

Use of Sizeof On a Pointer Type\Path 46:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1663
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c
Line	2348	2348
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c

Method wasm_get_module_mem_consumption(const WASMModule *module,

```
....
2348.      mem_conspn->data_segs_size = sizeof(WASMDataSeg*)
```

Use of Sizeof On a Pointer Type\Path 47:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1664
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c
Line	2390	2390
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c

Method wasm_get_module_inst_mem_consumption(const WASMModuleInstance *module_inst,

```
....
2390.       mem_conspn->memories_size = sizeof(WASMMemoryInstance *)
```

Use of Sizeof On a Pointer Type\Path 48:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1665
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c
Line	2404	2404
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-48105-FP.c

Method wasm_get_module_inst_mem_consumption(const WASMModuleInstance *module_inst,

```
....
2404.       mem_conspn->tables_size = sizeof(WASMTableInstance *)
```

Use of Sizeof On a Pointer Type\Path 49:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1666
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	546	546
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c

Method load_type_section(const uint8 *buf, const uint8 *buf_end, WASMModule *module,

```
....
546.          total_size = sizeof(WASMType*) * (uint64)type_count;
```

Use of Sizeof On a Pointer Type\Path 50:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1667
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	1874	1874
Object	sizeof	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c

Method load_function_section(const uint8 *buf, const uint8 *buf_end,

```
....
1874.          total_size = sizeof(WASMFunction*) * (uint64)func_count;
```

Improper Resource Access Authorization

Query Path:

CPP\Cx\CPP Low Visibility\Improper Resource Access Authorization Version:1

Categories

FISMA 2014: Identification And Authentication
NIST SP 800-53: AC-3 Access Enforcement (P1)
OWASP Top 10 2017: A2-Broken Authentication

Description

Improper Resource Access Authorization\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1281
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Line	449	449

Object	fgets	fgets

Code Snippet
File Name bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read (FILE *fp)

```
....
449.    while (fgets (buf, sizeof (buf), fp) != NULL)
```

Improper Resource Access Authorization\Path 2:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1282
Status New

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Line	465	465
Object	fgets	fgets

Code Snippet
File Name bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read_large (FILE *fp)

```
....
465.    TEST_VERIFY (fgets (buf, sizeof (buf), fp) != NULL);
```

Improper Resource Access Authorization\Path 3:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1283
Status New

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Line	449	449
Object	fgets	fgets

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read (FILE *fp)

```
....
449.    while (fgets (buf, sizeof (buf), fp) != NULL)
```

Improper Resource Access Authorization\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1284
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Line	465	465
Object	fgets	fgets

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read_large (FILE *fp)

```
....
465.    TEST_VERIFY (fgets (buf, sizeof (buf), fp) != NULL);
```

Improper Resource Access Authorization\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1285
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Line	449	449
Object	fgets	fgets

Code Snippet
File Name bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read (FILE *fp)

```
....
449.    while (fgets (buf, sizeof (buf), fp) != NULL)
```

Improper Resource Access Authorization\Path 6:

Severity	Low

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1286
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Line	465	465
Object	fgets	fgets

Code Snippet
File Name bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read_large (FILE *fp)

```
....
465.    TEST_VERIFY (fgets (buf, sizeof (buf), fp) != NULL);
```

Improper Resource Access Authorization\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1287
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Line	449	449
Object	fgets	fgets

Code Snippet
File Name bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read (FILE *fp)

```
....
449.    while (fgets (buf, sizeof (buf), fp) != NULL)
```

Improper Resource Access Authorization\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1288
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Line	465	465
Object	fgets	fgets

Code Snippet
File Name bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read_large (FILE *fp)

```
....
465.    TEST_VERIFY (fgets (buf, sizeof (buf), fp) != NULL);
```

Improper Resource Access Authorization\Path 9:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1289
Status New

	Source	Destination
File	c-ares@@c-ares-cares-1_16_0-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_16_0-CVE-2024-25629-TP.c
Line	49	49
Object	fgets	fgets

Code Snippet
File Name c-ares@@c-ares-cares-1_16_0-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Improper Resource Access Authorization\Path 10:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1290
Status New

	Source	Destination
File	c-ares@@c-ares-c-ares-1_17_0-CVE-2024-25629-TP.c	c-ares@@c-ares-c-ares-1_17_0-CVE-2024-25629-TP.c
Line	49	49

Object	fgets	fgets

Code Snippet
File Name c-ares@@c-ares-c-ares-1_17_0-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Improper Resource Access Authorization\Path 11:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1291
Status New

	Source	Destination
File	c-ares@@c-ares-cares-1_17_2-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_17_2-CVE-2024-25629-TP.c
Line	49	49
Object	fgets	fgets

Code Snippet
File Name c-ares@@c-ares-cares-1_17_2-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Improper Resource Access Authorization\Path 12:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1292
Status New

	Source	Destination
File	c-ares@@c-ares-cares-1_18_0-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_18_0-CVE-2024-25629-TP.c
Line	49	49
Object	fgets	fgets

Code Snippet
File Name c-ares@@c-ares-cares-1_18_0-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Improper Resource Access Authorization\Path 13:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1293
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_19_0-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_19_0-CVE-2024-25629-TP.c
Line	49	49
Object	fgets	fgets

Code Snippet
File Name c-ares@@c-ares-cares-1_19_0-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Improper Resource Access Authorization\Path 14:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1294
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_19_1-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_19_1-CVE-2024-25629-TP.c
Line	49	49
Object	fgets	fgets

Code Snippet
File Name c-ares@@c-ares-cares-1_19_1-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Improper Resource Access Authorization\Path 15:

Severity	Low

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1295
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_20_0-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_20_0-CVE-2024-25629-TP.c
Line	60	60
Object	fgets	fgets

Code Snippet
File Name c-ares@@c-ares-cares-1_20_0-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
60.         if (!fgets(*buf + offset, bytestoread, fp))
```

Improper Resource Access Authorization\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1296
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_26_0-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_26_0-CVE-2024-25629-TP.c
Line	58	58
Object	fgets	fgets

Code Snippet
File Name c-ares@@c-ares-cares-1_26_0-CVE-2024-25629-TP.c
Method ares_status_t ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
58.       if (!fgets(*buf + offset, bytestoread, fp)) {
```

Improper Resource Access Authorization\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1297
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Line	449	449
Object	buf	buf

Code Snippet
File Name bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read (FILE *fp)

```
....
449.    while (fgets (buf, sizeof (buf), fp) != NULL)
```

Improper Resource Access Authorization\Path 18:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1298
Status New

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Line	465	465
Object	buf	buf

Code Snippet
File Name bminor@@glibc-glibc-2.37.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read_large (FILE *fp)

```
....
465.    TEST_VERIFY (fgets (buf, sizeof (buf), fp) != NULL);
```

Improper Resource Access Authorization\Path 19:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1299
Status New

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Line	449	449

Object	buf	buf

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read (FILE *fp)

```
....
449.    while (fgets (buf, sizeof (buf), fp) != NULL)
```

Improper Resource Access Authorization\Path 20:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1300
Status New

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Line	465	465
Object	buf	buf

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read_large (FILE *fp)

```
....
465.    TEST_VERIFY (fgets (buf, sizeof (buf), fp) != NULL);
```

Improper Resource Access Authorization\Path 21:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1301
Status New

	Source	Destination
File	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Line	449	449
Object	buf	buf

Code Snippet
File Name bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read (FILE *fp)

```
....
449.    while (fgets (buf, sizeof (buf), fp) != NULL)
```

Improper Resource Access Authorization\Path 22:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1302
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Line	465	465
Object	buf	buf

Code Snippet
File Name bminor@@glibc-glibc-2.39.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read_large (FILE *fp)

```
....
465.    TEST_VERIFY (fgets (buf, sizeof (buf), fp) != NULL);
```

Improper Resource Access Authorization\Path 23:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1303
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Line	449	449
Object	buf	buf

Code Snippet
File Name bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read (FILE *fp)

```
....
449.    while (fgets (buf, sizeof (buf), fp) != NULL)
```

Improper Resource Access Authorization\Path 24:

Severity	Low

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1304
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Line	465	465
Object	buf	buf

Code Snippet
File Name bminor@@glibc-glibc-2.40.9000-CVE-2023-6246-FP.c
Method check_syslog_console_read_large (FILE *fp)

```
....
465.    TEST_VERIFY (fgets (buf, sizeof (buf), fp) != NULL);
```

Improper Resource Access Authorization\Path 25:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1305
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_16_0-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_16_0-CVE-2024-25629-TP.c
Line	49	49
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name c-ares@@c-ares-cares-1_16_0-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Improper Resource Access Authorization\Path 26:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1306
Status	New

	Source	Destination
File	c-ares@@c-ares-c-ares-1_17_0-CVE-2024-25629-TP.c	c-ares@@c-ares-c-ares-1_17_0-CVE-2024-25629-TP.c
Line	49	49
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name c-ares@@c-ares-c-ares-1_17_0-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Improper Resource Access Authorization\Path 27:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1307
Status New

	Source	Destination
File	c-ares@@c-ares-cares-1_17_2-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_17_2-CVE-2024-25629-TP.c
Line	49	49
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name c-ares@@c-ares-cares-1_17_2-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Improper Resource Access Authorization\Path 28:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1308
Status New

	Source	Destination
File	c-ares@@c-ares-cares-1_18_0-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_18_0-CVE-2024-25629-TP.c
Line	49	49

Object	BinaryExpr	BinaryExpr

Code Snippet
File Name c-ares@@c-ares-cares-1_18_0-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Improper Resource Access Authorization\Path 29:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1309
Status New

	Source	Destination
File	c-ares@@c-ares-cares-1_19_0-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_19_0-CVE-2024-25629-TP.c
Line	49	49
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name c-ares@@c-ares-cares-1_19_0-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Improper Resource Access Authorization\Path 30:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1310
Status New

	Source	Destination
File	c-ares@@c-ares-cares-1_19_1-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_19_1-CVE-2024-25629-TP.c
Line	49	49
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name c-ares@@c-ares-cares-1_19_1-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
49.         if (!fgets(*buf + offset, bytestoread, fp))
```

Improper Resource Access Authorization\Path 31:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1311
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_20_0-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_20_0-CVE-2024-25629-TP.c
Line	60	60
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name c-ares@@c-ares-cares-1_20_0-CVE-2024-25629-TP.c
Method int ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
60.         if (!fgets(*buf + offset, bytestoread, fp))
```

Improper Resource Access Authorization\Path 32:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1312
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_26_0-CVE-2024-25629-TP.c	c-ares@@c-ares-cares-1_26_0-CVE-2024-25629-TP.c
Line	58	58
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name c-ares@@c-ares-cares-1_26_0-CVE-2024-25629-TP.c
Method ares_status_t ares__read_line(FILE *fp, char **buf, size_t *bufsize)

```
....
58.       if (!fgets(*buf + offset, bytestoread, fp)) {
```

Improper Resource Access Authorization\Path 33:

Severity	Low

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1313
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	122	122
Object	token	token

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
122.          size_t read_num = fread(token, 1,
sizeof(sgx_launch_token_t), fp);
```

Improper Resource Access Authorization\Path 34:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1314
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	196	196
Object	buffer	buffer

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method read_file_to_buffer(const char *filename, uint32_t *ret_size)

```
....
196.      read_size = fread(buffer, 1, file_size, file);
```

Improper Resource Access Authorization\Path 35:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1315

Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	122	122
Object	token	token

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
122.          size_t read_num = fread(token, 1,
sizeof(sgx_launch_token_t), fp);
```

Improper Resource Access Authorization\Path 36:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1316
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	196	196
Object	buffer	buffer

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method read_file_to_buffer(const char *filename, uint32_t *ret_size)

```
....
196.      read_size = fread(buffer, 1, file_size, file);
```

Improper Resource Access Authorization\Path 37:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1317
Status	New

	Source	Destination

File	cesanta@@mongoose-newest-CVE-2020-8597-TP.c	cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Line	1366	1366
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Method static void eap_request(ppp_pcb *pcb, u_char *inp, int id, int len) {

```
....
1366.                           len = read(fd, rhostname + SRP_PSEUDO_LEN,
```

Improper Resource Access Authorization\Path 38:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1318
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	53	53
Object	path_buf	path_buf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method get_exe_path(char *path_buf, unsigned path_buf_size)

```
....
53.      ssize_t size = readlink("/proc/self/exe", path_buf,
path_buf_size - 1);
```

Improper Resource Access Authorization\Path 39:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1319
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	53	53

Object	path_buf	path_buf

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method get_exe_path(char *path_buf, unsigned path_buf_size)

```
....
53.      ssize_t size = readlink("/proc/self/exe", path_buf,
path_buf_size - 1);
```

Improper Resource Access Authorization\Path 40:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1320
Status New

	Source	Destination
File	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Line	463	463
Object	tmp	tmp

Code Snippet
File Name c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Method static int file_lookup(struct host_query *hquery)

```
....
463.                RegQueryValueEx(hkeyHosts, DATABASEPATH, NULL, NULL,
(LPBYTE)tmp,
```

Improper Resource Access Authorization\Path 41:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1321
Status New

	Source	Destination
File	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Line	464	464
Object	Address	Address

Code Snippet
File Name c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c

Method	static int file_lookup(struct host_query *hquery)

```
....
464.                                    &dwLength);
```

Improper Resource Access Authorization\Path 42:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1322
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Line	496	496
Object	fprintf	fprintf

Code Snippet

File Name	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Method	static int file_lookup(struct host_query *hquery)

```
....
496.            DEBUGF(fprintf(stderr, "fopen() failed with error: %d
%s\n", error,
```

Improper Resource Access Authorization\Path 43:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1323
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Line	498	498
Object	fprintf	fprintf

Code Snippet

File Name	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Method	static int file_lookup(struct host_query *hquery)

```
....
498.            DEBUGF(fprintf(stderr, "Error opening file: %s\n",
path_hosts));
```

Improper Resource Access Authorization\Path 44:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1324
Status	New

	Source	Destination
File	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Line	501	501
Object	fprintf	fprintf

Code Snippet
File Name c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
501.            DEBUGF(fprintf(stderr, "fopen() failed with error: %d
%s\n", error,
```

Improper Resource Access Authorization\Path 45:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1325
Status	New

	Source	Destination
File	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Line	503	503
Object	fprintf	fprintf

Code Snippet
File Name c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
503.            DEBUGF(fprintf(stderr, "Error opening file: %s\n",
path_hosts));
```

Improper Resource Access Authorization\Path 46:

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

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1326
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Line	495	495
Object	fprintf	fprintf

Code Snippet
File Name c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
495.            DEBUGF(fprintf(stderr, "fopen() failed with error: %d
%s\n", error,
```

Improper Resource Access Authorization\Path 47:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1327
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Line	497	497
Object	fprintf	fprintf

Code Snippet
File Name c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
497.            DEBUGF(fprintf(stderr, "Error opening file: %s\n",
path_hosts));
```

Improper Resource Access Authorization\Path 48:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1328
Status	New

	Source	Destination

File	c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c
Line	471	471
Object	fprintf	fprintf

Code Snippet
File Name c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
471.            DEBUGF(fprintf(stderr, "fopen() failed with error: %d
%s\n", error,
```

Improper Resource Access Authorization\Path 49:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1329
Status New

	Source	Destination
File	c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c
Line	473	473
Object	fprintf	fprintf

Code Snippet
File Name c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
473.            DEBUGF(fprintf(stderr, "Error opening file: %s\n",
path_hosts));
```

Improper Resource Access Authorization\Path 50:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1330
Status New

	Source	Destination
File	c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c
Line	498	498

Object	fprintf	fprintf

Code Snippet
File Name c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
498.            DEBUGF(fprintf(stderr, "fopen() failed with error: %d
%s\n", error,
```

Sizeof Pointer Argument

Query Path:
CPP\Cx\CPP Low Visibility\Sizeof Pointer Argument Version:0

Description

Sizeof Pointer Argument\Path 1:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2057
Status New

	Source	Destination
File	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Line	223	223
Object	buf	sizeof

Code Snippet
File Name c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Method static void test_unescape_char_quoted_one(const char *string, size_t n_string, bool escaped) {

```
....
223.          size_t n_out = sizeof(buf);
```

Sizeof Pointer Argument\Path 2:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2058
Status New

	Source	Destination
File	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Line	274	274

Object	buf	sizeof

Code Snippet
File Name c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Method static void test_unescape_char_unquoted_one(const char *string, size_t n_string, bool escaped) {

```
....
274.          size_t n_out = sizeof(buf);
```

Sizeof Pointer Argument\Path 3:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2059
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	498	498
Object	ecall_args	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method app_instance_main(void *wasm_module_inst, int app_argc, char **app_argv)

```
....
498.      if (app_argc + 2 > sizeof(ecall_args_buf) / sizeof(uint64_t))
{
```

Sizeof Pointer Argument\Path 4:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2060
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	535	535
Object	ecall_args	sizeof

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method	app_instance_func(void *wasm_module_inst, const char *func_name, int app_argc,

```
....
535.      if (app_argc + 3 > sizeof(ecall_args_buf) / sizeof(uint64_t))
{
```

Sizeof Pointer Argument\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2061
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	498	498
Object	ecall_args	sizeof

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method	app_instance_main(void *wasm_module_inst, int app_argc, char **app_argv)

```
....
498.      if (app_argc + 2 > sizeof(ecall_args_buf) / sizeof(uint64_t))
{
```

Sizeof Pointer Argument\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2062
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	535	535
Object	ecall_args	sizeof

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c

Method	app_instance_func(void *wasm_module_inst, const char *func_name, int app_argc,

```
....
535.      if (app_argc + 3 > sizeof(ecall_args_buf) / sizeof(uint64_t))
{
```

Sizeof Pointer Argument\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2063
Status	New

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	1735	1735
Object	buf	sizeof

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1735.      buf[sizeof(buf) - 1] = 0;
```

Sizeof Pointer Argument\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2064
Status	New

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	1735	1735
Object	buf	sizeof

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1735.      buf[sizeof(buf) - 1] = 0;
```

Sizeof Pointer Argument\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2065
Status	New

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	1640	1640
Object	buf	sizeof

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1640.       buf[sizeof(buf) - 1] = 0;
```

Sizeof Pointer Argument\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2066
Status	New

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	1724	1724
Object	buf	sizeof

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1724.       snprintf(buf, sizeof(buf), "LIST -L\r\n");
```

Sizeof Pointer Argument\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2067

Status	New

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	1733	1733
Object	buf	sizeof

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1733.       snprintf(buf, sizeof(buf), "LIST -L %s\r\n", filename);
```

Sizeof Pointer Argument\Path 12:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2068
Status	New

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	1724	1724
Object	buf	sizeof

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1724.       snprintf(buf, sizeof(buf), "LIST -L\r\n");
```

Sizeof Pointer Argument\Path 13:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2069
Status	New

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c

Line	1733	1733
Object	buf	sizeof

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1733.       snprintf(buf, sizeof(buf), "LIST -L %s\r\n", filename);
```

Sizeof Pointer Argument\Path 14:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2070
Status New

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	1629	1629
Object	buf	sizeof

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1629.       snprintf(buf, sizeof(buf), "LIST -L\r\n");
```

Sizeof Pointer Argument\Path 15:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2071
Status New

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	1638	1638
Object	buf	sizeof

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c

Method	xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1638.       snprintf(buf, sizeof(buf), "LIST -L %s\r\n", filename);
```

Sizeof Pointer Argument\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2072
Status	New

	Source	Destination
File	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Line	286	286
Object	buf	sizeof

Code Snippet	
File Name	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Method	static void test_unescape_char_unquoted_one(const char *string, size_t n_string, bool escaped) {

```
....
286.                    c_assert(n_out == sizeof(buf));
```

Sizeof Pointer Argument\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2073
Status	New

	Source	Destination
File	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Line	274	286
Object	buf	sizeof

Code Snippet	
File Name	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Method	static void test_unescape_char_unquoted_one(const char *string, size_t n_string, bool escaped) {

```
....
274.          size_t n_out = sizeof(buf);
....
286.                  c_assert(n_out == sizeof(buf));
```

Sizeof Pointer Argument\Path 18:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2074
Status	New

	Source	Destination
File	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Line	235	235
Object	buf	sizeof

Code Snippet
File Name c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Method static void test_unescape_char_quoted_one(const char *string, size_t n_string, bool escaped) {

```
....
235.                  c_assert(n_out == sizeof(buf) - 1);
```

Sizeof Pointer Argument\Path 19:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2075
Status	New

	Source	Destination
File	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Line	223	235
Object	buf	sizeof

Code Snippet
File Name c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Method static void test_unescape_char_quoted_one(const char *string, size_t n_string, bool escaped) {

```
....
223.          size_t n_out = sizeof(buf);
....
235.                  c_assert(n_out == sizeof(buf) - 1);
```

Sizeof Pointer Argument\Path 20:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2076
Status	New

	Source	Destination
File	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Line	239	239
Object	buf	sizeof

Code Snippet
File Name c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Method static void test_unescape_char_quoted_one(const char *string, size_t n_string, bool escaped) {

```
....
239.                  c_assert(n_out == sizeof(buf) - 2);
```

Sizeof Pointer Argument\Path 21:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2077
Status	New

	Source	Destination
File	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Line	223	239
Object	buf	sizeof

Code Snippet
File Name c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Method static void test_unescape_char_quoted_one(const char *string, size_t n_string, bool escaped) {

```
....
223.          size_t n_out = sizeof(buf);
....
239.                  c_assert(n_out == sizeof(buf) - 2);
```

Sizeof Pointer Argument\Path 22:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2078
Status	New

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	1782	1782
Object	buf	sizeof

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1782.        if ((len = recv(ctxt->dataFd, &buf[indx], sizeof(buf) -
(indx + 1), 0)) < 0) {
```

Sizeof Pointer Argument\Path 23:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2079
Status	New

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	1782	1782
Object	buf	sizeof

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1782.       if ((len = recv(ctxt->dataFd, &buf[indx], sizeof(buf) -
(indx + 1), 0)) < 0) {
```

Sizeof Pointer Argument\Path 24:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2080
Status	New

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	1681	1681
Object	buf	sizeof

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1681.       if ((len = recv(ctxt->dataFd, &buf[indx], sizeof(buf) -
(indx + 1), 0)) < 0) {
```

Sizeof Pointer Argument\Path 25:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2081
Status	New

	Source	Destination
File	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Line	292	292
Object	buf	sizeof

Code Snippet
File Name c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Method static void test_unescape_char_unquoted_one(const char *string, size_t n_string, bool escaped) {

```
....
292.                        c_assert(n_out == sizeof(buf));
```

Sizeof Pointer Argument\Path 26:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2082
Status	New

	Source	Destination
File	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Line	274	292
Object	buf	sizeof

Code Snippet
File Name c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Method static void test_unescape_char_unquoted_one(const char *string, size_t n_string, bool escaped) {

```
....
274.          size_t n_out = sizeof(buf);
....
292.                           c_assert(n_out == sizeof(buf));
```

Sizeof Pointer Argument\Path 27:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2083
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Line	193	193
Object	dir_list	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Method int main(int argc, char *argv[])

```
....
193.              if (dir_list_size >= sizeof(dir_list) / sizeof(char*))
{
```

Sizeof Pointer Argument\Path 28:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2084
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	500	500
Object	ecall_args	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method app_instance_main(void *wasm_module_inst, int app_argc, char **app_argv)

```
....
500.                    (uint64_t *)malloc(sizeof(uint64_t) * (app_argc
+ 2)))) {
```

Sizeof Pointer Argument\Path 29:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2085
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	498	500
Object	ecall_args	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method app_instance_main(void *wasm_module_inst, int app_argc, char **app_argv)

```
....
498.      if (app_argc + 2 > sizeof(ecall_args_buf) / sizeof(uint64_t))
{
....
500.                    (uint64_t *)malloc(sizeof(uint64_t) * (app_argc
+ 2)))) {
```

Sizeof Pointer Argument\Path 30:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2086
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	537	537
Object	ecall_args	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method app_instance_func(void *wasm_module_inst, const char *func_name, int app_argc,

```
....
537.                    (uint64_t *)malloc(sizeof(uint64_t) * (app_argc
+ 3)))) {
```

Sizeof Pointer Argument\Path 31:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2087
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	535	537
Object	ecall_args	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method app_instance_func(void *wasm_module_inst, const char *func_name, int app_argc,

```
....
535.      if (app_argc + 3 > sizeof(ecall_args_buf) / sizeof(uint64_t))
{
....
537.                    (uint64_t *)malloc(sizeof(uint64_t) * (app_argc
+ 3)))) {
```

Sizeof Pointer Argument\Path 32:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2088
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	500	500
Object	ecall_args	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method app_instance_main(void *wasm_module_inst, int app_argc, char **app_argv)

```
....
500.                    (uint64_t *)malloc(sizeof(uint64_t) * (app_argc
+ 2)))) {
```

Sizeof Pointer Argument\Path 33:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2089
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	498	500
Object	ecall_args	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method app_instance_main(void *wasm_module_inst, int app_argc, char **app_argv)

```
....
498.      if (app_argc + 2 > sizeof(ecall_args_buf) / sizeof(uint64_t))
{
....
500.                    (uint64_t *)malloc(sizeof(uint64_t) * (app_argc
+ 2)))) {
```

Sizeof Pointer Argument\Path 34:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2090
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	537	537
Object	ecall_args	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method app_instance_func(void *wasm_module_inst, const char *func_name, int app_argc,

```
....
537.                    (uint64_t *)malloc(sizeof(uint64_t) * (app_argc
+ 3)))) {
```

Sizeof Pointer Argument\Path 35:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2091
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	535	537
Object	ecall_args	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method app_instance_func(void *wasm_module_inst, const char *func_name, int app_argc,

```
....
535.      if (app_argc + 3 > sizeof(ecall_args_buf) / sizeof(uint64_t))
{
....
537.                    (uint64_t *)malloc(sizeof(uint64_t) * (app_argc
+ 3)))) {
```

Sizeof Pointer Argument\Path 36:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2092
Status	New

	Source	Destination
File	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Line	297	297
Object	buf	sizeof

Code Snippet
File Name c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Method static void test_unescape_char_unquoted_one(const char *string, size_t n_string, bool escaped) {

```
....
297.                            c_assert(n_out == sizeof(buf) - 1);
```

Sizeof Pointer Argument\Path 37:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2093
Status	New

	Source	Destination
File	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp	c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Line	274	297
Object	buf	sizeof

Code Snippet
File Name c-util@@c-shquote-v1.0.0-CVE-2022-31212-FP.cpp
Method static void test_unescape_char_unquoted_one(const char *string, size_t n_string, bool escaped) {

```
....
274.          size_t n_out = sizeof(buf);
....
297.                            c_assert(n_out == sizeof(buf) - 1);
```

Sizeof Pointer Argument\Path 38:

Severity	Low
Result State	To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2094
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Line	205	205
Object	env_list	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c

Method int main(int argc, char *argv[])

```
....
205.              if (env_list_size >= sizeof(env_list) / sizeof(char*))
{
```

Sizeof Pointer Argument\Path 39:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2095
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Line	274	274
Object	dir_list	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c

Method main(int argc, char *argv[])

```
....
274.              if (dir_list_size >= sizeof(dir_list) / sizeof(char
*)) {
```

Sizeof Pointer Argument\Path 40:

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

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2096
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c
Line	207	207
Object	dir_list	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c

Method int main(int argc, char *argv[])

```
....
207.              if (dir_list_size >= sizeof(dir_list) / sizeof(char*))
{
```

Sizeof Pointer Argument\Path 41:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2097
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c
Line	274	274
Object	dir_list	sizeof

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c

Method main(int argc, char *argv[])

```
....
274.              if (dir_list_size >= sizeof(dir_list) / sizeof(char
*)) {
```

Sizeof Pointer Argument\Path 42:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&

	pathid=2098
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	737	737
Object	dir_list	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
737.                if (dir_list_size >= sizeof(dir_list) / sizeof(char
*)) {
```

Sizeof Pointer Argument\Path 43:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2099
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	737	737
Object	dir_list	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
737.                if (dir_list_size >= sizeof(dir_list) / sizeof(char
*)) {
```

Sizeof Pointer Argument\Path 44:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2100
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Line	286	286
Object	env_list	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
286.              if (env_list_size >= sizeof(env_list) / sizeof(char
*)) {
```

Sizeof Pointer Argument\Path 45:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2101
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c
Line	219	219
Object	env_list	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-48105-FP.c
Method int main(int argc, char *argv[])

```
....
219.              if (env_list_size >= sizeof(env_list) / sizeof(char*))
{
```

Sizeof Pointer Argument\Path 46:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2102
Status New

	Source	Destination

File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c
Line	286	286
Object	env_list	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
286.              if (env_list_size >= sizeof(env_list) / sizeof(char
*)) {
```

Sizeof Pointer Argument\Path 47:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2103
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	749	749
Object	env_list	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
749.              if (env_list_size >= sizeof(env_list) / sizeof(char
*)) {
```

Sizeof Pointer Argument\Path 48:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2104
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-

	FP.c	FP.c
Line	749	749
Object	env_list	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
749.              if (env_list_size >= sizeof(env_list) / sizeof(char
*)) {
```

Sizeof Pointer Argument\Path 49:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2105
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Line	195	195
Object	dir_list	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Method int main(int argc, char *argv[])

```
....
195.                              (int)(sizeof(dir_list) /
sizeof(char*)));
```

Sizeof Pointer Argument\Path 50:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2106
Status New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c

Line	193	195
Object	dir_list	sizeof

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-02-27-2020-CVE-2023-48105-FP.c
Method int main(int argc, char *argv[])

```
....
193.              if (dir_list_size >= sizeof(dir_list) / sizeof(char*))
{
....
195.                                  (int)(sizeof(dir_list) /
sizeof(char*)));
```

TOCTOU

Query Path:
CPP\Cx\CPP Low Visibility\TOCTOU Version:1

Description

TOCTOU\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2939
Status	New

The enclave_init method in bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	114	114
Object	fopen	fopen

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
114.      fp = fopen(token_path, "rb");
```

TOCTOU\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2940

Status	New

The enclave_init method in bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	115	115
Object	fopen	fopen

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
115.      if (fp == NULL && (fp = fopen(token_path, "wb")) == NULL) {
```

TOCTOU\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2941
Status	New

The read_file_to_buffer method in bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	181	181
Object	fopen	fopen

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method read_file_to_buffer(const char *filename, uint32_t *ret_size)

```
....
181.      if (!(file = fopen(filename, "r"))) {
```

TOCTOU\Path 4:

Severity	Low
Result State	To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2942
Status	New

The dump_pgo_prof_data method in bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	652	652
Object	fopen	fopen

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method dump_pgo_prof_data(void *module_inst, const char *path)

```
....
652.      if (!(file = fopen(path, "wb"))) {
```

TOCTOU\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2943
Status	New

The enclave_init method in bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	114	114
Object	fopen	fopen

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
114.      fp = fopen(token_path, "rb");
```

TOCTOU\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2944
Status	New

The enclave_init method in bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	115	115
Object	fopen	fopen

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
115.      if (fp == NULL && (fp = fopen(token_path, "wb")) == NULL) {
```

TOCTOU\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2945
Status	New

The read_file_to_buffer method in bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	181	181
Object	fopen	fopen

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method read_file_to_buffer(const char *filename, uint32_t *ret_size)

```
....
181.      if (!(file = fopen(filename, "r"))) {
```

TOCTOU\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2946
Status	New

The dump_pgo_prof_data method in bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	652	652
Object	fopen	fopen

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method dump_pgo_prof_data(void *module_inst, const char *path)

```
....
652.      if (!(file = fopen(path, "wb"))) {
```

TOCTOU\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2947
Status	New

The file_lookup method in c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Line	486	486
Object	fopen	fopen

Code Snippet

File Name	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Method	static int file_lookup(struct host_query *hquery)

```
....
486.    fp = fopen(path_hosts, "r");
```

TOCTOU\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2948
Status	New

The file_lookup method in c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Line	491	491
Object	fopen	fopen

Code Snippet

File Name	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Method	static int file_lookup(struct host_query *hquery)

```
....
491.    fp = fopen(path_hosts, "r");
```

TOCTOU\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2949
Status	New

The file_lookup method in c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Line	485	485
Object	fopen	fopen

Code Snippet
File Name c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
485.    fp = fopen(path_hosts, "r");
```

TOCTOU\Path 12:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2950
Status	New

The file_lookup method in c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c
Line	461	461
Object	fopen	fopen

Code Snippet
File Name c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
461.    fp = fopen(path_hosts, "r");
```

TOCTOU\Path 13:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2951
Status	New

The file_lookup method in c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c
Line	487	487
Object	fopen	fopen

Code Snippet	
File Name	c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c
Method	static int file_lookup(struct host_query *hquery)

```
....
487.    fp = fopen(path_hosts, "r");
```

TOCTOU\Path 14:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2952
Status	New

The file_lookup method in c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c
Line	491	491
Object	fopen	fopen

Code Snippet	
File Name	c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c
Method	static int file_lookup(struct host_query *hquery)

```
....
491.    fp = fopen(path_hosts, "r");
```

TOCTOU\Path 15:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2953
Status	New

The file_lookup method in c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c
Line	506	506

Object	fopen	fopen

Code Snippet
File Name c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
506.    fp = fopen(path_hosts, "r");
```

TOCTOU\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2954
Status	New

The main method in chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	2078	2078
Object	fopen	fopen

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method int main(int argc, char **argv) {

```
....
2078.       output = fopen("/tmp/tstdata", "w");
```

TOCTOU\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2955
Status	New

The main method in chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c

Line	2078	2078
Object	fopen	fopen

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method int main(int argc, char **argv) {

```
....
2078.       output = fopen("/tmp/tstdata", "w");
```

TOCTOU\Path 18:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2956
Status	New

The main method in chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	1962	1962
Object	fopen	fopen

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method int main(int argc, char **argv) {

```
....
1962.       output = fopen("/tmp/tstdata", "w");
```

TOCTOU\Path 19:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2957
Status	New

The open_pn_file method in cesanta@@mongoose-newest-CVE-2020-8597-TP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	cesanta@@mongoose-newest-CVE-	cesanta@@mongoose-newest-CVE-

	2020-8597-TP.c	2020-8597-TP.c
Line	1231	1231
Object	open	open

Code Snippet
File Name cesanta@@mongoose-newest-CVE-2020-8597-TP.c
Method open_pn_file(modebits)

```
....
1231.       fd = open(path, modebits, S_IRUSR | S_IWUSR);
```

TOCTOU\Path 20:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2958
Status	New

The ProcessProperties method in chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Line	172	172
Object	open	open

Code Snippet
File Name chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
172.      s_tty_fd = open("/dev/tty", O_WRONLY);
```

TOCTOU\Path 21:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2959
Status	New

The ProcessProperties method in chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination

File	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Line	212	212
Object	open	open

Code Snippet
File Name chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
212.    int fd0 = open(getenv("PS_STDIN"), O_RDONLY);
```

TOCTOU\Path 22:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2960
Status New

The ProcessProperties method in chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Line	215	215
Object	open	open

Code Snippet
File Name chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
215.    int fd1 = open(getenv("PS_STDOUT"), O_WRONLY);
```

TOCTOU\Path 23:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2961
Status New

The ProcessProperties method in chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Line	218	218
Object	open	open

Code Snippet
File Name chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
218.    int fd2 = open(getenv("PS_STDERR"), O_WRONLY);
```

TOCTOU\Path 24:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2962
Status New

The MessageHandlerInput method in chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c	chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Line	280	280
Object	open	open

Code Snippet
File Name chromium@@chromium-108.0.5351.1-CVE-2021-44109-FP.c
Method void MessageHandlerInput(struct PP_Var key,

```
....
280.    int fd = open(filename, O_RDONLY);
```

TOCTOU\Path 25:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2963
Status New

The ProcessProperties method in chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Line	172	172
Object	open	open

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
172.      s_tty_fd = open("/dev/tty", O_WRONLY);
```

TOCTOU\Path 26:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2964
Status New

The ProcessProperties method in chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Line	212	212
Object	open	open

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
212.    int fd0 = open(getenv("PS_STDIN"), O_RDONLY);
```

TOCTOU\Path 27:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2965
Status New

The ProcessProperties method in chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Line	215	215
Object	open	open

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
215.    int fd1 = open(getenv("PS_STDOUT"), O_WRONLY);
```

TOCTOU\Path 28:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2966
Status New

The ProcessProperties method in chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Line	218	218
Object	open	open

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
218.    int fd2 = open(getenv("PS_STDERR"), O_WRONLY);
```

TOCTOU\Path 29:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2967
Status New

The MessageHandlerInput method in chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Line	280	280
Object	open	open

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-44109-FP.c
Method void MessageHandlerInput(struct PP_Var key,

```
....
280.    int fd = open(filename, O_RDONLY);
```

TOCTOU\Path 30:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2968
Status New

The ProcessProperties method in chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Line	172	172
Object	open	open

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
172.      s_tty_fd = open("/dev/tty", O_WRONLY);
```

TOCTOU\Path 31:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2969
Status New

The ProcessProperties method in chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Line	212	212
Object	open	open

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
212.    int fd0 = open(getenv("PS_STDIN"), O_RDONLY);
```

TOCTOU\Path 32:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2970
Status New

The ProcessProperties method in chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Line	215	215
Object	open	open

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
215.    int fd1 = open(getenv("PS_STDOUT"), O_WRONLY);
```

TOCTOU\Path 33:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2971
Status New

The ProcessProperties method in chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Line	218	218
Object	open	open

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
218.    int fd2 = open(getenv("PS_STDERR"), O_WRONLY);
```

TOCTOU\Path 34:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2972
Status New

The MessageHandlerInput method in chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Line	280	280
Object	open	open

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-44109-FP.c
Method void MessageHandlerInput(struct PP_Var key,

```
....
280.    int fd = open(filename, O_RDONLY);
```

TOCTOU\Path 35:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2973
Status New

The ProcessProperties method in chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Line	172	172
Object	open	open

Code Snippet
File Name chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
172.      s_tty_fd = open("/dev/tty", O_WRONLY);
```

TOCTOU\Path 36:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2974
Status New

The ProcessProperties method in chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Line	212	212
Object	open	open

Code Snippet
File Name chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
212.    int fd0 = open(getenv("PS_STDIN"), O_RDONLY);
```

TOCTOU\Path 37:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2975
Status New

The ProcessProperties method in chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Line	215	215
Object	open	open

Code Snippet
File Name chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
215.    int fd1 = open(getenv("PS_STDOUT"), O_WRONLY);
```

TOCTOU\Path 38:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2976
Status New

The ProcessProperties method in chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Line	218	218
Object	open	open

Code Snippet
File Name chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
218.    int fd2 = open(getenv("PS_STDERR"), O_WRONLY);
```

TOCTOU\Path 39:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2977
Status New

The MessageHandlerInput method in chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c	chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Line	280	280
Object	open	open

Code Snippet
File Name chromium@@chromium-117.0.5881.1-CVE-2021-44109-FP.c
Method void MessageHandlerInput(struct PP_Var key,

```
....
280.    int fd = open(filename, O_RDONLY);
```

TOCTOU\Path 40:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2978
Status New

The ProcessProperties method in chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Line	172	172
Object	open	open

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
172.      s_tty_fd = open("/dev/tty", O_WRONLY);
```

TOCTOU\Path 41:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2979
Status New

The ProcessProperties method in chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Line	212	212
Object	open	open

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
212.    int fd0 = open(getenv("PS_STDIN"), O_RDONLY);
```

TOCTOU\Path 42:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2980
Status New

The ProcessProperties method in chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Line	215	215
Object	open	open

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
215.    int fd1 = open(getenv("PS_STDOUT"), O_WRONLY);
```

TOCTOU\Path 43:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2981
Status New

The ProcessProperties method in chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Line	218	218
Object	open	open

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Method int ProcessProperties(void) {

```
....
218.    int fd2 = open(getenv("PS_STDERR"), O_WRONLY);
```

TOCTOU\Path 44:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=2982
Status	New

The MessageHandlerInput method in chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Line	280	280
Object	open	open

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-44109-FP.c
Method void MessageHandlerInput(struct PP_Var key,

```
....
280.    int fd = open(filename, O_RDONLY);
```

Potential Off by One Error in Loops

Query Path:
CPP\Cx\CPP Heuristic\Potential Off by One Error in Loops Version:1

Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.1 - Injection flaws - particularly SQL injection
NIST SP 800-53: SI-16 Memory Protection (P1)
OWASP Top 10 2017: A1-Injection

Description
Potential Off by One Error in Loops\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=797
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c at line 5682 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	6057	6057
Object	<=	<=

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c

Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
6057.                   for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=798
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c at line 3223 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c
Line	3338	3338
Object	<=	<=

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-01-29-2021-CVE-2023-52284-FP.c

Method wasm_loader_find_block_addr(BlockAddr *block_addr_cache,

```
....
3338.                   for (i = 0; i <= count; i++) /* lableidxs */
```

Potential Off by One Error in Loops\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=799
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c at line 6444 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	6837	6837
Object	<=	<=

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c

Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
6837.                   for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=800
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c at line 3756 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Line	3880	3880
Object	<=	<=

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-48105-FP.c
Method	wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
3880.                   for (i = 0; i <= count; i++) /* lableidxs */
```

Potential Off by One Error in Loops\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=801
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c at line 6444 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	6837	6837
Object	<=	<=

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method	wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
6837.                   for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=802
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c at line 3756 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination

File	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Line	3880	3880
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-05-18-2022-CVE-2023-52284-FP.c
Method wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
3880.                   for (i = 0; i <= count; i++) /* lableidxs */
```

Potential Off by One Error in Loops\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=803
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c at line 4787 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	5023	5023
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
5023.                   for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=804
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c at line 2910 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Line	3013	3013
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-06-15-2020-CVE-2023-52284-FP.c
Method wasm_loader_find_block_addr(BlockAddr *block_addr_cache,

```
....
3013.                   for (i = 0; i <= count; i++) /* lableidxs */
```

Potential Off by One Error in Loops\Path 9:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=805
Status New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c at line 3609 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	3724	3724
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Method wasm_loader_find_block_addr(BlockAddr *block_addr_cache,

```
....
3724.                   for (i = 0; i <= count; i++) /* lableidxs */
```

Potential Off by One Error in Loops\Path 10:

Severity Low
Result State To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=806
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c at line 6137 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c
Line	6514	6514
Object	<=	<=

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-08-10-2021-CVE-2023-52284-FP.c

Method wasm_loader_prepare_bytecode(WASMModule *module,

```
....
6514.                   for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=807
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c at line 5441 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Line	5818	5818
Object	<=	<=

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c

Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
5818.                   for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 12:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=808
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c at line 3187 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c
Line	3302	3302
Object	<=	<=

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-09-29-2020-CVE-2023-52284-FP.c

Method wasm_loader_find_block_addr(BlockAddr *block_addr_cache,

```
....
3302.                   for (i = 0; i <= count; i++) /* lableidxs */
```

Potential Off by One Error in Loops\Path 13:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=809
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c at line 6625 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Line	7028	7028
Object	<=	<=

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Method	wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
7028.                    for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 14:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=810
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c at line 3900 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Line	4024	4024
Object	<=	<=

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-48105-FP.c
Method	wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
4024.                    for (i = 0; i <= count; i++) /* lableidxs */
```

Potential Off by One Error in Loops\Path 15:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=811
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c at line 6625 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-

	FP.c	FP.c
Line	7028	7028
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
7028.                   for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=812
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c at line 3900 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Line	4024	4024
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.0.0-CVE-2023-52284-FP.c
Method wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
4024.                   for (i = 0; i <= count; i++) /* lableidxs */
```

Potential Off by One Error in Loops\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=813
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c at line 6845 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Line	7247	7247
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
7247.                   for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 18:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=814
Status New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c at line 4108 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Line	4232	4232
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-48105-FP.c
Method wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
4232.                   for (i = 0; i <= count; i++) /* lableidxs */
```

Potential Off by One Error in Loops\Path 19:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=815
Status New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c at line 6845 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Line	7247	7247
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
7247.                   for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 20:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=816
Status New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c at line 4108 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Line	4232	4232
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.1.2-CVE-2023-52284-FP.c
Method wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
4232.                   for (i = 0; i <= count; i++) /* lableidxs */
```

Potential Off by One Error in Loops\Path 21:

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

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=817
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c at line 7197 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Line	7599	7599
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
7599.                   for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 22:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=818
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c at line 4454 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Line	4578	4578
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-48105-FP.c
Method wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
4578.                   for (i = 0; i <= count; i++) /* lableidxs */
```

Potential Off by One Error in Loops\Path 23:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=819
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c at line 7197 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c
Line	7599	7599
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
7599.                    for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 24:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=820
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c at line 4454 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c
Line	4578	4578
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.0-CVE-2023-52284-FP.c
Method wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
4578.                   for (i = 0; i <= count; i++) /* lableidxs */
```

Potential Off by One Error in Loops\Path 25:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=821
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c at line 7199 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c
Line	7601	7601
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
7601.                   for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 26:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=822
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c at line 4455 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c
Line	4579	4579
Object	<=	<=

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-48105-TP.c
Method	wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
4579.                    for (i = 0; i <= count; i++) /* lableidxs */
```

Potential Off by One Error in Loops\Path 27:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=823
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c at line 7199 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c
Line	7601	7601
Object	<=	<=

Code Snippet	
File Name	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c
Method	wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
7601.                    for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 28:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=824
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c at line 4455 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c

Line	4579	4579
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.2.3-CVE-2023-52284-TP.c
Method wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
4579.                   for (i = 0; i <= count; i++) /* lableidxs */
```

Potential Off by One Error in Loops\Path 29:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=825
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c at line 6318 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c
Line	6698	6698
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
6698.                   for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 30:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=826
Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c at line 3724 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c
Line	3847	3847
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-48105-FP.c
Method wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
3847.                   for (i = 0; i <= count; i++)      /* lableidxs */
```

Potential Off by One Error in Loops\Path 31:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=827
Status New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c at line 6318 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c
Line	6698	6698
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c
Method wasm_loader_prepare_bytecode(WASMModule *module, WASMFunction *func,

```
....
6698.                   for (i = 0; i <= count; i++) {
```

Potential Off by One Error in Loops\Path 32:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=828

Status	New

The buffer allocated by <= in bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c at line 3724 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c
Line	3847	3847
Object	<=	<=

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-12-30-2021-CVE-2023-52284-FP.c
Method wasm_loader_find_block_addr(WASMExecEnv *exec_env, BlockAddr *block_addr_cache,

```
....
3847.                   for (i = 0; i <= count; i++)      /* lableidxs */
```

Incorrect Permission Assignment For Critical Resources

Query Path:
CPP\Cx\CPP Low Visibility\Incorrect Permission Assignment For Critical Resources Version:1

Categories

FISMA 2014: Access Control
NIST SP 800-53: AC-3 Access Enforcement (P1)
OWASP Top 10 2017: A2-Broken Authentication

Description
Incorrect Permission Assignment For Critical Resources\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1361
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	114	114
Object	fp	fp

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
114.      fp = fopen(token_path, "rb");
```

Incorrect Permission Assignment For Critical Resources\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1362
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	115	115
Object	fp	fp

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
115.      if (fp == NULL && (fp = fopen(token_path, "wb")) == NULL) {
```

Incorrect Permission Assignment For Critical Resources\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1363
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	181	181
Object	file	file

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method read_file_to_buffer(const char *filename, uint32_t *ret_size)

```
....
181.      if (!(file = fopen(filename, "r"))) {
```

Incorrect Permission Assignment For Critical Resources\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1364
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	652	652
Object	file	file

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method dump_pgo_prof_data(void *module_inst, const char *path)

```
....
652.      if (!(file = fopen(path, "wb"))) {
```

Incorrect Permission Assignment For Critical Resources\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1365
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	114	114
Object	fp	fp

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
114.      fp = fopen(token_path, "rb");
```

Incorrect Permission Assignment For Critical Resources\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&

	pathid=1366
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	115	115
Object	fp	fp

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method enclave_init(sgx_enclave_id_t *p_eid)

```
....
115.      if (fp == NULL && (fp = fopen(token_path, "wb")) == NULL) {
```

Incorrect Permission Assignment For Critical Resources\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1367
Status	New

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	181	181
Object	file	file

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method read_file_to_buffer(const char *filename, uint32_t *ret_size)

```
....
181.      if (!(file = fopen(filename, "r"))) {
```

Incorrect Permission Assignment For Critical Resources\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1368
Status	New

	Source	Destination

File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	652	652
Object	file	file

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method dump_pgo_prof_data(void *module_inst, const char *path)

```
....
652.      if (!(file = fopen(path, "wb"))) {
```

Incorrect Permission Assignment For Critical Resources\Path 9:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1369
Status New

	Source	Destination
File	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Line	486	486
Object	fp	fp

Code Snippet
File Name c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Method static int file_lookup(struct host_query *hquery)

```
....
486.    fp = fopen(path_hosts, "r");
```

Incorrect Permission Assignment For Critical Resources\Path 10:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1370
Status New

	Source	Destination
File	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Line	491	491
Object	fp	fp

Code Snippet	
File Name	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Method	static int file_lookup(struct host_query *hquery)

```
....
491.    fp = fopen(path_hosts, "r");
```

Incorrect Permission Assignment For Critical Resources\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1371
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Line	485	485
Object	fp	fp

Code Snippet	
File Name	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Method	static int file_lookup(struct host_query *hquery)

```
....
485.    fp = fopen(path_hosts, "r");
```

Incorrect Permission Assignment For Critical Resources\Path 12:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1372
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c
Line	461	461
Object	fp	fp

Code Snippet	
File Name	c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c
Method	static int file_lookup(struct host_query *hquery)

```
....
461.    fp = fopen(path_hosts, "r");
```

Incorrect Permission Assignment For Critical Resources\Path 13:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1373
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c
Line	487	487
Object	fp	fp

Code Snippet
File Name c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
487.    fp = fopen(path_hosts, "r");
```

Incorrect Permission Assignment For Critical Resources\Path 14:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1374
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c
Line	491	491
Object	fp	fp

Code Snippet
File Name c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
491.    fp = fopen(path_hosts, "r");
```

Incorrect Permission Assignment For Critical Resources\Path 15:

Severity	Low

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1375
Status	New

	Source	Destination
File	c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c
Line	506	506
Object	fp	fp

Code Snippet
File Name c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
506.    fp = fopen(path_hosts, "r");
```

Incorrect Permission Assignment For Critical Resources\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1376
Status	New

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	2078	2078
Object	output	output

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method int main(int argc, char **argv) {

```
....
2078.       output = fopen("/tmp/tstdata", "w");
```

Incorrect Permission Assignment For Critical Resources\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1377
Status	New

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	2078	2078
Object	output	output

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method int main(int argc, char **argv) {

```
....
2078.       output = fopen("/tmp/tstdata", "w");
```

Incorrect Permission Assignment For Critical Resources\Path 18:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1378
Status New

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	1962	1962
Object	output	output

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method int main(int argc, char **argv) {

```
....
1962.       output = fopen("/tmp/tstdata", "w");
```

Arithmenic Operation On Boolean

Query Path:
CPP\Cx\CPP Low Visibility\Arithmenic Operation On Boolean Version:1

Categories

FISMA 2014: Audit And Accountability
NIST SP 800-53: SC-5 Denial of Service Protection (P1)

Description

Arithmenic Operation On Boolean\Path 1:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=837

Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c
Line	181	181
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
181.                    SYSLOG_HEADER (pri, timestamp, &msgoff, pid));
```

Arithmenic Operation On Boolean\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=838
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c
Line	181	181
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
181.                    SYSLOG_HEADER (pri, timestamp, &msgoff, pid));
```

Arithmenic Operation On Boolean\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=839
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c

Line	214	214
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
214.                    SYSLOG_HEADER (pri, timestamp, &msgoff, pid));
```

Arithmenic Operation On Boolean\Path 4:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=840
Status New

	Source	Destination
File	bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c
Line	214	214
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
214.                    SYSLOG_HEADER (pri, timestamp, &msgoff, pid));
```

Arithmenic Operation On Boolean\Path 5:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=841
Status New

	Source	Destination
File	bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c	bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c
Line	231	231
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name bminor@@glibc-glibc-2.36.9000-CVE-2023-6246-FP.c

Method	__vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
231.                "\n" + (buf[bufsize - 1] == '\n'));
```

Arithmenic Operation On Boolean\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=842
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Line	181	181
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
181.                    SYSLOG_HEADER (pri, timestamp, &msgoff, pid));
```

Arithmenic Operation On Boolean\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=843
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Line	181	181
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
181.                    SYSLOG_HEADER (pri, timestamp, &msgoff, pid));
```

Arithmenic Operation On Boolean\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=844
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Line	212	212
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
212.                    SYSLOG_HEADER (pri, timestamp, &msgoff, pid));
```

Arithmenic Operation On Boolean\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=845
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Line	212	212
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
212.                    SYSLOG_HEADER (pri, timestamp, &msgoff, pid));
```

Arithmenic Operation On Boolean\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=846
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Line	235	235
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name bminor@@glibc-glibc-2.37.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
235.                "\n" + (buf[bufsize - 1] == '\n'));
```

Arithmenic Operation On Boolean\Path 11:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=847
Status New

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Line	183	183
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
183.                    SYSLOG_HEADER (pri, timestamp, &msgoff, pid));
```

Arithmenic Operation On Boolean\Path 12:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=848
Status New

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Line	183	183

Object	BinaryExpr	BinaryExpr

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
183.                    SYSLOG_HEADER (pri, timestamp, &msgoff, pid));
```

Arithmenic Operation On Boolean\Path 13:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=849
Status New

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Line	214	214
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
214.                    SYSLOG_HEADER (pri, timestamp, &msgoff, pid));
```

Arithmenic Operation On Boolean\Path 14:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=850
Status New

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Line	214	214
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
214.                    SYSLOG_HEADER (pri, timestamp, &msgoff, pid));
```

Arithmenic Operation On Boolean\Path 15:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=851
Status	New

	Source	Destination
File	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c	bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Line	237	237
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name bminor@@glibc-glibc-2.38.9000-CVE-2023-6779-FP.c
Method __vsyslog_internal (int pri, const char *fmt, va_list ap,

```
....
237.                "\n" + (buf[bufsize - 1] == '\n'));
```

Exposure of System Data to Unauthorized Control Sphere

Query Path:
CPP\Cx\CPP Low Visibility\Exposure of System Data to Unauthorized Control Sphere Version:1

Categories

FISMA 2014: Configuration Management
NIST SP 800-53: AC-3 Access Enforcement (P1)

Description

Exposure of System Data to Unauthorized Control Sphere\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1379
Status	New

The system data read by xmlNanoFTPCloseConnection in the file chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c at line 1529 is potentially exposed by xmlNanoFTPCloseConnection found in chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c at line 1529.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	1547	1547

Object	perror	perror

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPCloseConnection(void *ctx) {

```
....
1547.       perror("select");
```

Exposure of System Data to Unauthorized Control Sphere\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1380
Status	New

The system data read by xmlNanoFTPList in the file chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c at line 1708 is potentially exposed by xmlNanoFTPList found in chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c at line 1708.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	1762	1762
Object	perror	perror

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1762.           perror("select");
```

Exposure of System Data to Unauthorized Control Sphere\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1381
Status	New

The system data read by xmlNanoFTPGet in the file chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c at line 1883 is potentially exposed by xmlNanoFTPGet found in chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c at line 1883.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c

Line	1907	1907
Object	perror	perror

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPGet(void *ctx, ftpDataCallback callback, void *userData,

```
....
1907.           perror("select");
```

Exposure of System Data to Unauthorized Control Sphere\Path 4:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1382
Status New

The system data read by xmlNanoFTPCloseConnection in the file chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c at line 1529 is potentially exposed by xmlNanoFTPCloseConnection found in chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c at line 1529.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	1547	1547
Object	perror	perror

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPCloseConnection(void *ctx) {

```
....
1547.       perror("select");
```

Exposure of System Data to Unauthorized Control Sphere\Path 5:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1383
Status New

The system data read by xmlNanoFTPList in the file chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c at line 1708 is potentially exposed by xmlNanoFTPList found in chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c at line 1708.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-	chromium@@chromium-114.0.5707.0-

	CVE-2021-3520-FP.c	CVE-2021-3520-FP.c
Line	1762	1762
Object	perror	perror

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPList(void *ctx, ftpListCallback callback, void *userData,

```
....
1762.           perror("select");
```

Exposure of System Data to Unauthorized Control Sphere\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1384
Status	New

The system data read by xmlNanoFTPGet in the file chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c at line 1883 is potentially exposed by xmlNanoFTPGet found in chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c at line 1883.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	1907	1907
Object	perror	perror

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPGet(void *ctx, ftpDataCallback callback, void *userData,

```
....
1907.           perror("select");
```

Exposure of System Data to Unauthorized Control Sphere\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1385
Status	New

The system data read by file_lookup in the file c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c at line 432 is potentially exposed by file_lookup found in c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c at line 432.

	Source	Destination

File	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Line	441	498
Object	getenv	fprintf

Code Snippet
File Name c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Method static int file_lookup(struct host_query *hquery)

```
....
441.         path_hosts = getenv("CARES_HOSTS");
....
498.             DEBUGF(fprintf(stderr, "Error opening file: %s\n",
path_hosts));
```

Exposure of System Data to Unauthorized Control Sphere\Path 8:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1386
Status New

The system data read by file_lookup in the file c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c at line 437 is potentially exposed by file_lookup found in c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c at line 437.

	Source	Destination
File	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Line	446	503
Object	getenv	fprintf

Code Snippet
File Name c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
446.         path_hosts = getenv("CARES_HOSTS");
....
503.             DEBUGF(fprintf(stderr, "Error opening file: %s\n",
path_hosts));
```

Exposure of System Data to Unauthorized Control Sphere\Path 9:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1387
Status New

The system data read by file_lookup in the file c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c at line 431 is potentially exposed by file_lookup found in c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c at line 431.

	Source	Destination
File	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Line	440	497
Object	getenv	fprintf

Code Snippet
File Name c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
440.        path_hosts = getenv("CARES_HOSTS");
....
497.            DEBUGF(fprintf(stderr, "Error opening file: %s\n",
path_hosts));
```

Exposure of System Data to Unauthorized Control Sphere\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1388
Status	New

The system data read by file_lookup in the file c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c at line 407 is potentially exposed by file_lookup found in c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c at line 407.

	Source	Destination
File	c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c
Line	416	473
Object	getenv	fprintf

Code Snippet
File Name c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
416.        path_hosts = getenv("CARES_HOSTS");
....
473.            DEBUGF(fprintf(stderr, "Error opening file: %s\n",
path_hosts));
```

Exposure of System Data to Unauthorized Control Sphere\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1389
Status	New

The system data read by file_lookup in the file c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c at line 428 is potentially exposed by file_lookup found in c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c at line 428.

	Source	Destination
File	c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c
Line	437	500
Object	getenv	fprintf

Code Snippet
File Name c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
437.         path_hosts = getenv("CARES_HOSTS");
....
500.             DEBUGF(fprintf(stderr, "Error opening file: %s\n",
path_hosts));
```

Exposure of System Data to Unauthorized Control Sphere\Path 12:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1390
Status	New

The system data read by file_lookup in the file c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c at line 428 is potentially exposed by file_lookup found in c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c at line 428.

	Source	Destination
File	c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c
Line	437	504
Object	getenv	fprintf

Code Snippet
File Name c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
437.        path_hosts = ares_strdup(getenv("CARES_HOSTS"));
....
504.            DEBUGF(fprintf(stderr, "Error opening file: %s\n",
path_hosts));
```

Exposure of System Data to Unauthorized Control Sphere\Path 13:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1391
Status	New

The system data read by file_lookup in the file c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c at line 443 is potentially exposed by file_lookup found in c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c at line 443.

	Source	Destination
File	c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c
Line	452	519
Object	getenv	fprintf

Code Snippet
File Name c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
452.        path_hosts = ares_strdup(getenv("CARES_HOSTS"));
....
519.            DEBUGF(fprintf(stderr, "Error opening file: %s\n",
path_hosts));
```

Potential Path Traversal

Query Path:
CPP\Cx\CPP Low Visibility\Potential Path Traversal Version:0

Categories

OWASP Top 10 2013: A4-Insecure Direct Object References
OWASP Top 10 2017: A5-Broken Access Control

Description

Potential Path Traversal\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1392
Status	New

Method main at line 667 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c gets user input from the argv element. This element's value then flows through the code and is eventually used in a file path for local disk access in read_file_to_buffer at line 170 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c. This may cause a Path Traversal vulnerability.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	667	181
Object	argv	filename

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
667.  main(int argc, char *argv[])
```

File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method read_file_to_buffer(const char *filename, uint32_t *ret_size)

```
....
181.      if (!(file = fopen(filename, "r"))) {
```

Potential Path Traversal\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1393
Status	New

Method main at line 667 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c gets user input from the argv element. This element's value then flows through the code and is eventually used in a file path for local disk access in dump_pgo_prof_data at line 610 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c. This may cause a Path Traversal vulnerability.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	667	652
Object	argv	path

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
667.  main(int argc, char *argv[])
```

File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c

Method dump_pgo_prof_data(void *module_inst, const char *path)

```
....
652.      if (!(file = fopen(path, "wb"))) {
```

Potential Path Traversal\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1394
Status	New

Method main at line 667 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c gets user input from the argv element. This element's value then flows through the code and is eventually used in a file path for local disk access in read_file_to_buffer at line 170 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c. This may cause a Path Traversal vulnerability.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	667	181
Object	argv	filename

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c

Method main(int argc, char *argv[])

```
....
667.  main(int argc, char *argv[])
```

File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c

Method read_file_to_buffer(const char *filename, uint32_t *ret_size)

```
....
181.      if (!(file = fopen(filename, "r"))) {
```

Potential Path Traversal\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&

	pathid=1395
Status	New

Method main at line 667 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c gets user input from the argv element. This element's value then flows through the code and is eventually used in a file path for local disk access in dump_pgo_prof_data at line 610 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c. This may cause a Path Traversal vulnerability.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	667	652
Object	argv	path

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c

Method main(int argc, char *argv[])

```
....
667.  main(int argc, char *argv[])
```

File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c

Method dump_pgo_prof_data(void *module_inst, const char *path)

```
....
652.      if (!(file = fopen(path, "wb"))) {
```

Potential Path Traversal\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1396
Status	New

Method file_lookup at line 432 of c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c gets user input from the getenv element. This element's value then flows through the code and is eventually used in a file path for local disk access in file_lookup at line 432 of c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c. This may cause a Path Traversal vulnerability.

	Source	Destination
File	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c	c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c
Line	441	486
Object	getenv	path_hosts

Code Snippet

File Name c-ares@@c-ares-cares-1_16_0-CVE-2020-14354-TP.c

Method	static int file_lookup(struct host_query *hquery)

```
....
441.        path_hosts = getenv("CARES_HOSTS");
....
486.    fp = fopen(path_hosts, "r");
```

Potential Path Traversal\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1397
Status	New

Method file_lookup at line 437 of c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c gets user input from the getenv element. This element's value then flows through the code and is eventually used in a file path for local disk access in file_lookup at line 437 of c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c. This may cause a Path Traversal vulnerability.

	Source	Destination
File	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Line	446	491
Object	getenv	path_hosts

Code Snippet	
File Name	c-ares@@c-ares-c-ares-1_17_0-CVE-2020-14354-FP.c
Method	static int file_lookup(struct host_query *hquery)

```
....
446.        path_hosts = getenv("CARES_HOSTS");
....
491.    fp = fopen(path_hosts, "r");
```

Potential Path Traversal\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1398
Status	New

Method file_lookup at line 431 of c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c gets user input from the getenv element. This element's value then flows through the code and is eventually used in a file path for local disk access in file_lookup at line 431 of c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c. This may cause a Path Traversal vulnerability.

	Source	Destination
File	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Line	440	485

Object	getenv	path_hosts

Code Snippet
File Name c-ares@@c-ares-cares-1_17_2-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
440.         path_hosts = getenv("CARES_HOSTS");
....
485.     fp = fopen(path_hosts, "r");
```

Potential Path Traversal\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1399
Status	New

Method file_lookup at line 407 of c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c gets user input from the getenv element. This element's value then flows through the code and is eventually used in a file path for local disk access in file_lookup at line 407 of c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c. This may cause a Path Traversal vulnerability.

	Source	Destination
File	c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c
Line	416	461
Object	getenv	path_hosts

Code Snippet
File Name c-ares@@c-ares-cares-1_18_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
416.         path_hosts = getenv("CARES_HOSTS");
....
461.     fp = fopen(path_hosts, "r");
```

Potential Path Traversal\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1400
Status	New

Method file_lookup at line 428 of c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c gets user input from the getenv element. This element's value then flows through the code and is eventually used in a file path for local disk access in file_lookup at line 428 of c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c. This may cause a Path Traversal vulnerability.

	Source	Destination
File	c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c
Line	437	487
Object	getenv	path_hosts

Code Snippet
File Name c-ares@@c-ares-cares-1_19_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
437.        path_hosts = getenv("CARES_HOSTS");
....
487.    fp = fopen(path_hosts, "r");
```

Potential Path Traversal\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1401
Status	New

Method file_lookup at line 428 of c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c gets user input from the getenv element. This element's value then flows through the code and is eventually used in a file path for local disk access in file_lookup at line 428 of c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c. This may cause a Path Traversal vulnerability.

	Source	Destination
File	c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c
Line	437	491
Object	getenv	path_hosts

Code Snippet
File Name c-ares@@c-ares-cares-1_19_1-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
437.        path_hosts = ares_strdup(getenv("CARES_HOSTS"));
....
491.    fp = fopen(path_hosts, "r");
```

Potential Path Traversal\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1402
Status	New

Method file_lookup at line 443 of c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c gets user input from the getenv element. This element's value then flows through the code and is eventually used in a file path for local disk access in file_lookup at line 443 of c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c. This may cause a Path Traversal vulnerability.

	Source	Destination
File	c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c	c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c
Line	452	506
Object	getenv	path_hosts

Code Snippet
File Name c-ares@@c-ares-cares-1_20_0-CVE-2020-14354-FP.c
Method static int file_lookup(struct host_query *hquery)

```
....
452.         path_hosts = ares_strdup(getenv("CARES_HOSTS"));
....
506.     fp = fopen(path_hosts, "r");
```

Heuristic Buffer Overflow malloc

Query Path:
CPP\Cx\CPP Heuristic\Heuristic Buffer Overflow malloc Version:0

Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.2 - Buffer overflows
NIST SP 800-53: SI-10 Information Input Validation (P1)
OWASP Top 10 2017: A1-Injection

Description
Heuristic Buffer Overflow malloc\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=829
Status	New

The size of the buffer used by app_instance_main in app_argc, at line 492 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argc, at line 667 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	667	500
Object	argc	app_argc

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
667.  main(int argc, char *argv[])
```

File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method app_instance_main(void *wasm_module_inst, int app_argc, char **app_argv)

```
....
500.                    (uint64_t *)malloc(sizeof(uint64_t) * (app_argc
+ 2)))) {
```

Heuristic Buffer Overflow malloc\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=830
Status	New

The size of the buffer used by app_instance_main in BinaryExpr, at line 492 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argc, at line 667 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	667	500
Object	argc	BinaryExpr

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
667.  main(int argc, char *argv[])
```

File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method app_instance_main(void *wasm_module_inst, int app_argc, char **app_argv)

```
....
500.                    (uint64_t *)malloc(sizeof(uint64_t) * (app_argc
+ 2)))) {
```

Heuristic Buffer Overflow malloc\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=831
Status	New

The size of the buffer used by app_instance_main in BinaryExpr, at line 492 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argc, at line 667 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	667	500
Object	argc	BinaryExpr

Code Snippet

File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c

Method main(int argc, char *argv[])

```
....
667.  main(int argc, char *argv[])
```

File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c

Method app_instance_main(void *wasm_module_inst, int app_argc, char **app_argv)

```
....
500.                    (uint64_t *)malloc(sizeof(uint64_t) * (app_argc
+ 2)))) {
```

Heuristic Buffer Overflow malloc\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=832
Status	New

The size of the buffer used by read_file_to_buffer in file_size, at line 170 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 667 of bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination

File	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Line	667	190
Object	argv	file_size

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
667.  main(int argc, char *argv[])
```

File Name bytecodealliance@@wasm-micro-runtime-WAMR-1.3.0-CVE-2023-48105-FP.c
Method read_file_to_buffer(const char *filename, uint32_t *ret_size)

```
....
190.      if (!(buffer = (unsigned char *)malloc(file_size))) {
```

Heuristic Buffer Overflow malloc\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=833
Status	New

The size of the buffer used by app_instance_main in app_argc, at line 492 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argc, at line 667 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	667	500
Object	argc	app_argc

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
667.  main(int argc, char *argv[])
```

File Name	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method	app_instance_main(void *wasm_module_inst, int app_argc, char **app_argv)

```
....
500.                    (uint64_t *)malloc(sizeof(uint64_t) * (app_argc
+ 2)))) {
```

Heuristic Buffer Overflow malloc\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=834
Status	New

The size of the buffer used by app_instance_main in BinaryExpr, at line 492 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argc, at line 667 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	667	500
Object	argc	BinaryExpr

Code Snippet

File Name	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method	main(int argc, char *argv[])

```
....
667.  main(int argc, char *argv[])
```

File Name	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method	app_instance_main(void *wasm_module_inst, int app_argc, char **app_argv)

```
....
500.                    (uint64_t *)malloc(sizeof(uint64_t) * (app_argc
+ 2)))) {
```

Heuristic Buffer Overflow malloc\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=835
Status	New

The size of the buffer used by app_instance_main in BinaryExpr, at line 492 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argc, at line 667 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	667	500
Object	argc	BinaryExpr

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method main(int argc, char *argv[])

```
....
667.  main(int argc, char *argv[])
```

File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method app_instance_main(void *wasm_module_inst, int app_argc, char **app_argv)

```
....
500.                    (uint64_t *)malloc(sizeof(uint64_t) * (app_argc
+ 2)))) {
```

Heuristic Buffer Overflow malloc\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=836
Status	New

The size of the buffer used by read_file_to_buffer in file_size, at line 170 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to argv, at line 667 of bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c, to overwrite the target buffer.

	Source	Destination
File	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Line	667	190
Object	argv	file_size

Code Snippet
File Name bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c

Method	main(int argc, char *argv[])

```
....
667.  main(int argc, char *argv[])
```

File Name	bytecodealliance@@wasm-micro-runtime-WAMR-2.0.0-CVE-2023-48105-FP.c
Method	read_file_to_buffer(const char *filename, uint32_t *ret_size)

```
....
190.      if (!(buffer = (unsigned char *)malloc(file_size))) {
```

Reliance on DNS Lookups in a Decision

Query Path:
CPP\Cx\CPP Low Visibility\Reliance on DNS Lookups in a Decision Version:0

Categories

FISMA 2014: Identification And Authentication
NIST SP 800-53: SC-23 Session Authenticity (P1)

Description

Reliance on DNS Lookups in a Decision\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1795
Status	New

The xmlNanoFTPConnect method performs a reverse DNS lookup with getaddrinfo, at line 832 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c. The application then makes a security decision, !=, in chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c line 832, even though this hostname is not reliable and can be easily spoofed.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	866	866
Object	getaddrinfo	!=

Code Snippet

File Name	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method	xmlNanoFTPConnect(void *ctx) {

```
....
866.            if (getaddrinfo (proxy, NULL, &hints, &result) != 0) {
```

Reliance on DNS Lookups in a Decision\Path 2:

Severity	Low
Result State	To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1796
Status	New

The xmlNanoFTPConnect method performs a reverse DNS lookup with getaddrinfo, at line 832 of chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c. The application then makes a security decision, !=, in chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c line 832, even though this hostname is not reliable and can be easily spoofed.

	Source	Destination
File	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c	chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Line	872	872
Object	getaddrinfo	!=

Code Snippet
File Name chromium@@chromium-111.0.5530.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
872.            if (getaddrinfo (ctxt->hostname, NULL, &hints, &result)
!= 0) {
```

Reliance on DNS Lookups in a Decision\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1797
Status	New

The xmlNanoFTPConnect method performs a reverse DNS lookup with getaddrinfo, at line 832 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c. The application then makes a security decision, !=, in chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c line 832, even though this hostname is not reliable and can be easily spoofed.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	866	866
Object	getaddrinfo	!=

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
866.            if (getaddrinfo (proxy, NULL, &hints, &result) != 0) {
```

Reliance on DNS Lookups in a Decision\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1798
Status	New

The xmlNanoFTPConnect method performs a reverse DNS lookup with getaddrinfo, at line 832 of chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c. The application then makes a security decision, !=, in chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c line 832, even though this hostname is not reliable and can be easily spoofed.

	Source	Destination
File	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c	chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Line	872	872
Object	getaddrinfo	!=

Code Snippet
File Name chromium@@chromium-114.0.5707.0-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
872.            if (getaddrinfo (ctxt->hostname, NULL, &hints, &result)
!= 0) {
```

Reliance on DNS Lookups in a Decision\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1799
Status	New

The xmlNanoFTPConnect method performs a reverse DNS lookup with getaddrinfo, at line 771 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c. The application then makes a security decision, !=, in chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c line 771, even though this hostname is not reliable and can be easily spoofed.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	805	805
Object	getaddrinfo	!=

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
805.            if (getaddrinfo (proxy, NULL, &hints, &result) != 0) {
```

Reliance on DNS Lookups in a Decision\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000009&projectid=5&pathid=1800
Status	New

The xmlNanoFTPConnect method performs a reverse DNS lookup with getaddrinfo, at line 771 of chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c. The application then makes a security decision, !=, in chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c line 771, even though this hostname is not reliable and can be easily spoofed.

	Source	Destination
File	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c	chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Line	811	811
Object	getaddrinfo	!=

Code Snippet
File Name chromium@@chromium-119.0.6045.17-CVE-2021-3520-FP.c
Method xmlNanoFTPConnect(void *ctx) {

```
....
811.            if (getaddrinfo (ctxt->hostname, NULL, &hints, &result)
!= 0) {
```

Buffer Overflow Indexes

Risk

What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

Cause

How does it happen

Buffer Overflows can manifest in numerous different variations. In it's most basic form, the attack controls a buffer, which is then copied to a smaller buffer without size verification. Because the attacker's source buffer is larger than the program's target buffer, the attacker's data overwrites whatever is next on the stack, allowing the attacker to control program structures.
Alternatively, the vulnerability could be the result of improper bounds checking; exposing internal memory addresses outside of their valid scope; allowing the attacker to control the size of the target buffer; or various other forms.

General Recommendations

How to avoid it

- Always perform proper bounds checking before copying buffers or strings.
- Prefer to use safer functions and structures, e.g. safe string classes over `char*`, `strncpy` over `strcpy`, and so on.
- Consistently apply tests for the size of buffers.
- Do not return variable addresses outside the scope of their variables.

Source Code Examples

Buffer Overflow boundedcpy

Risk

What might happen

Allowing tainted inputs to set the size of how many bytes to copy from source to destination may cause memory corruption, unexpected behavior, instability and data leakage. In some cases, such as when additional and specific areas of memory are also controlled by user input, it may result in code execution.

Cause

How does it happen

Should the size of the amount of bytes to copy from source to destination be greater than the size of the destination, an overflow will occur, and memory beyond the intended buffer will get overwritten. Since this size value is derived from user input, the user may provide an invalid and dangerous buffer size.

General Recommendations

How to avoid it

- Do not trust memory allocation sizes provided by the user; derive them from the copied values instead.
- If memory allocation by a provided value is absolutely required, restrict this size to safe values only. Specifically ensure that this value does not exceed the destination buffer's size.

Source Code Examples

CPP

Size Parameter is Influenced by User Input

```
char dest_buf[10];
memset(dest_buf, '\0', sizeof(dest_buf));
strncpy(dest_buf, src_buf, size); //Assuming size is provided by user input
```

Validating Destination Buffer Length

```
char dest_buf[10];
memset(dest_buf, '\0', sizeof(dest_buf));
if (size < sizeof(dest_buf) && sizeof(src_buf) >= size) //Assuming size is provided by user
input
{
      strncpy(dest_buf, src_buf, size);
}
else
{
      //...
}
```

Buffer Overflow IndexFromInput

Risk

What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

Cause

How does it happen

Buffer Overflows can manifest in numerous different variations. In it's most basic form, the attack controls a buffer, which is then copied to a smaller buffer without size verification. Because the attacker's source buffer is larger than the program's target buffer, the attacker's data overwrites whatever is next on the stack, allowing the attacker to control program structures.
Alternatively, the vulnerability could be the result of improper bounds checking; exposing internal memory addresses outside of their valid scope; allowing the attacker to control the size of the target buffer; or various other forms.

General Recommendations

How to avoid it

- Always perform proper bounds checking before copying buffers or strings.
- Prefer to use safer functions and structures, e.g. safe string classes over `char*`, `strncpy` over `strcpy`, and so on.
- Consistently apply tests for the size of buffers.
- Do not return variable addresses outside the scope of their variables.

Source Code Examples

Format String Attack

Risk

What might happen

In environments with unmanaged memory, allowing attackers to control format strings could enable them to access areas of memory to which they should not have access, including reading other restricted variables, misrepresenting data, and possibly even overwriting unauthorized areas of memory. It is even possible this could further lead to buffer overflows and arbitrary code execution under certain circumstance.

Cause

How does it happen

The application allows user input to influence the string argument used for formatted print functions. This family of functions expects the first argument to designate the relative format of dynamically constructed output string, including how to represent each of the other arguments.

Allowing an external user or attacker to control this string, allows them to control the functioning of the printing function, and thus to access unexpected areas of memory.

General Recommendations

How to avoid it

Generic Guidance:

- Do not allow user input or any other external data to influence the format strings.
- Ensure that all string format functions are called with a static string as the format parameter, and that the correct number of arguments are passed to the function, according to the static format string.
- Alternatively, validate all user input before using it in the format string parameter to print format functions, and ensure formatting tokens are not included in the input.

Specific Recommendations:

- Do not include user input directly in the format string parameter (often the first or second argument) to formatting functions.
- Alternatively, use controlled information derived from the input, such as size or length, in the format string - but not the actual contents of the input itself.

Source Code Examples

CPP

Dynamic Formatting String - First Parameter of printf

```
printf("Hello, ");
printf(name); // If name contains tokens, it could retrieve arbitrary values from memory or
```

```
cause a crash
```

Static Formatting String - First Parameter of printf is Static

```
printf("Hello, %s", name);
```

Buffer Overflow StrcpyStrcat

Risk

What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

Cause

How does it happen

Buffer Overflows can manifest in numerous different variations. In it's most basic form, the attack controls a buffer, which is then copied to a smaller buffer without size verification. Because the attacker's source buffer is larger than the program's target buffer, the attacker's data overwrites whatever is next on the stack, allowing the attacker to control program structures.
Alternatively, the vulnerability could be the result of improper bounds checking; exposing internal memory addresses outside of their valid scope; allowing the attacker to control the size of the target buffer; or various other forms.

General Recommendations

How to avoid it

- Always perform proper bounds checking before copying buffers or strings.
- Prefer to use safer functions and structures, e.g. safe string classes over `char*`, `strncpy` over `strcpy`, and so on.
- Consistently apply tests for the size of buffers.
- Do not return variable addresses outside the scope of their variables.

Source Code Examples

Improper Null Termination

Weakness ID: 170 *(Weakness Base)* **Status:** Incomplete

Description

Description Summary

The software does not terminate or incorrectly terminates a string or array with a null character or equivalent terminator.

Extended Description

Null termination errors frequently occur in two different ways. An off-by-one error could cause a null to be written out of bounds, leading to an overflow. Or, a program could use a strncpy() function call incorrectly, which prevents a null terminator from being added at all. Other scenarios are possible.

Time of Introduction

- Implementation

Applicable Platforms

Languages

C

C++

Platform Notes

Conceptually, this does not just apply to the C language; any language or representation that involves a terminator could have this type of problem.

Common Consequences

Scope	Effect
Confidentiality Integrity	The case of an omitted null character is the most dangerous of the possible issues. This will almost certainly result in information disclosure, and possibly a buffer overflow condition, which may be exploited to execute arbitrary code.
Confidentiality Integrity Availability	If a null character is omitted from a string, then most string-copying functions will read data until they locate a null character, even outside of the intended boundaries of the string. This could: • cause a crash due to a segmentation fault • cause sensitive adjacent memory to be copied and sent to an outsider • trigger a buffer overflow when the copy is bering written to a fixed-size buffer
Integrity Availability	Misplaced null characters may result in any number of security problems. The biggest issue is a subset of buffer overflow, and write-what-where conditions, where data corruption occurs from the writing of a null character over valid data, or even instructions. A randomly placed null character may put the system into an undefined state, and therefore make it prone to crashing. A misplaced null character may corrupt other data in memory
Access Control	Should the null character corrupt the process flow, or affect a flag controlling access, it may lead to logical errors which allow for the execution of arbitrary code.

Likelihood of Exploit

Medium

Demonstrative Examples

Example 1

The following code reads from cfgfile and copies the input into inputbuf using strcpy(). The code mistakenly assumes that inputbuf will always contain a NULL terminator.

(Bad Code)

Example Language: **C**

```
#define MAXLEN 1024
...
char *pathbuf[MAXLEN];
...
read(cfgfile,inputbuf,MAXLEN); //does not null terminate
strcpy(pathbuf,input_buf); //requires null terminated input
...
```

The code above will behave correctly if the data read from cfgfile is null terminated on disk as expected. But if an attacker is able to modify this input so that it does not contain the expected NULL character, the call to strcpy() will continue copying from memory until it encounters an arbitrary NULL character. This will likely overflow the destination buffer and, if the attacker can control the contents of memory immediately following inputbuf, can leave the application susceptible to a buffer overflow attack.

Example 2

In the following code, readlink() expands the name of a symbolic link stored in the buffer path so that the buffer filename contains the absolute path of the file referenced by the symbolic link. The length of the resulting value is then calculated using strlen().

(Bad Code)

Example Language: **C**

```
char buf[MAXPATH];
...
readlink(path, buf, MAXPATH);
int length = strlen(filename);
...
```

The code above will not behave correctly because the value read into buf by readlink() will not be null terminated. In testing, vulnerabilities like this one might not be caught because the unused contents of buf and the memory immediately following it may be NULL, thereby causing strlen() to appear as if it is behaving correctly. However, in the wild strlen() will continue traversing memory until it encounters an arbitrary NULL character on the stack, which results in a value of length that is much larger than the size of buf and may cause a buffer overflow in subsequent uses of this value. Buffer overflows aside, whenever a single call to readlink() returns the same value that has been passed to its third argument, it is impossible to know whether the name is precisely that many bytes long, or whether readlink() has truncated the name to avoid overrunning the buffer. Traditionally, strings are represented as a region of memory containing data terminated with a NULL character. Older string-handling methods frequently rely on this NULL character to determine the length of the string. If a buffer that does not contain a NULL terminator is passed to one of these functions, the function will read past the end of the buffer. Malicious users typically exploit this type of vulnerability by injecting data with unexpected size or content into the application. They may provide the malicious input either directly as input to the program or indirectly by modifying application resources, such as configuration files. In the event that an attacker causes the application to read beyond the bounds of a buffer, the attacker may be able use a resulting buffer overflow to inject and execute arbitrary code on the system.

Example 3

While the following example is not exploitable, it provides a good example of how nulls can be omitted or misplaced, even when "safe" functions are used:

(Bad Code)

Example Language: **C**

```
#include <stdio.h>
#include <string.h>

int main() {

char longString[] = "String signifying nothing";
char shortString[16];

strncpy(shortString, longString, 16);
printf("The last character in shortString is: %c %1$x\n", shortString[15]);
return (0);
}
```

The above code gives the following output: The last character in shortString is: l 6c So, the shortString array does not end in a NULL character, even though the "safe" string function strncpy() was used.

Observed Examples

Reference	Description
CVE-2000-0312	Attacker does not null-terminate argv[] when invoking another program.
CVE-2003-0777	Interrupted step causes resultant lack of null termination.
CVE-2004-1072	Fault causes resultant lack of null termination, leading to buffer expansion.
CVE-2001-1389	Multiple vulnerabilities related to improper null termination.
CVE-2003-0143	Product does not null terminate a message buffer after snprintf-like call, leading to overflow.

Potential Mitigations

Phase: Requirements

Use a language that is not susceptible to these issues. However, be careful of null byte interaction errors (CWE-626) with lower-level constructs that may be written in a language that is susceptible.

Phase: Implementation

Ensure that all string functions used are understood fully as to how they append null characters. Also, be wary of off-by-one errors when appending nulls to the end of strings.

Phase: Implementation

If performance constraints permit, special code can be added that validates null-termination of string buffers, this is a rather naive and error-prone solution.

Phase: Implementation

Switch to bounded string manipulation functions. Inspect buffer lengths involved in the buffer overrun trace reported with the defect.

Phase: Implementation

Add code that fills buffers with nulls (however, the length of buffers still needs to be inspected, to ensure that the non null-terminated string is not written at the physical end of the buffer).

Weakness Ordinalities

Ordinality	Description
Resultant	*(where the weakness is typically related to the presence of some other weaknesses)*

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	20	Improper Input Validation	**Seven Pernicious Kingdoms (primary)700**
ChildOf	Category	169	Technology-Specific	**Development**

			Special Elements	**Concepts (primary)699**
ChildOf	Weakness Class	707	Improper Enforcement of Message or Data Structure	**Research Concepts (primary)1000**
ChildOf	Category	730	OWASP Top Ten 2004 Category A9 - Denial of Service	**Weaknesses in OWASP Top Ten (2004) (primary)711**
ChildOf	Category	741	CERT C Secure Coding Section 07 - Characters and Strings (STR)	**Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734**
ChildOf	Category	748	CERT C Secure Coding Section 50 - POSIX (POS)	Weaknesses Addressed by the CERT C Secure Coding Standard734
CanPrecede	Weakness Base	120	Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')	Research Concepts1000
CanPrecede	Weakness Variant	126	Buffer Over-read	Research Concepts1000
PeerOf	Weakness Base	463	Deletion of Data Structure Sentinel	Research Concepts1000
PeerOf	Weakness Base	464	Addition of Data Structure Sentinel	Research Concepts1000
CanAlsoBe	Weakness Variant	147	Improper Neutralization of Input Terminators	Research Concepts1000
MemberOf	View	630	Weaknesses Examined by SAMATE	**Weaknesses Examined by SAMATE (primary)630**
CanFollow	Weakness Base	193	Off-by-one Error	Research Concepts1000
CanFollow	Weakness Class	682	Incorrect Calculation	Research Concepts1000

Relationship Notes

Factors: this is usually resultant from other weaknesses such as off-by-one errors, but it can be primary to boundary condition violations such as buffer overflows. In buffer overflows, it can act as an expander for assumed-immutable data.

Overlaps missing input terminator.

f Causal Nature

Explicit

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
PLOVER			Improper Null Termination
7 Pernicious Kingdoms			String Termination Error
CLASP			Miscalculated null termination
OWASP Top Ten 2004	A9	CWE More Specific	Denial of Service
CERT C Secure Coding	POS30-C		Use the readlink() function properly
CERT C Secure Coding	STR03-C		Do not inadvertently truncate a null-terminated byte string
CERT C Secure Coding	STR32-C		Null-terminate byte strings as required

White Box Definitions

A weakness where the code path has:

1. end statement that passes a data item to a null-terminated string function
2. start statement that produces the improper null-terminated data item

Where "produces" is defined through the following scenarios:

1. data item never ended with null-terminator
2. null-terminator is re-written

Maintenance Notes

As currently described, this entry is more like a category than a weakness.

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	PLOVER		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-01		KDM Analytics	External
	added/updated white box definitions		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Causal Nature, Common Consequences, Description, Likelihood of Exploit, Maintenance Notes, Relationships, Other Notes, Relationship Notes, Taxonomy Mappings, Weakness Ordinalities		
2008-11-24	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Common Consequences		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2009-07-17	KDM Analytics		External
	Improved the White Box Definition		
2009-07-27	CWE Content Team	MITRE	Internal
	updated Common Consequences, Other Notes, Potential Mitigations, White Box Definitions		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Description		

BACK TO TOP

Buffer Overflow AddressOfLocalVarReturned

Risk

What might happen

A use after free error will cause code to use an area of memory previously assigned with a specific value, which has since been freed and may have been overwritten by another value. This error will likely cause unexpected behavior, memory corruption and crash errors. In some cases where the freed and used section of memory is used to determine execution flow, and the error can be induced by an attacker, this may result in execution of malicious code.

Cause

How does it happen

Pointers to variables allow code to have an address with a set size to a dynamically allocated variable. Eventually, the pointer's destination may become free - either explicitly in code, such as when programmatically freeing this variable, or implicitly, such as when a local variable is returned - once it is returned, the variable's scope is released. Once freed, this memory will be re-used by the application, overwritten with new data. At this point, dereferencing this pointer will potentially resolve newly written and unexpected data.

General Recommendations

How to avoid it

- Do not return local variables or pointers
- Review code to ensure no flow allows use of a pointer after it has been explicitly freed

---

Source Code Examples

Buffer Overflow boundcpy WrongSizeParam

Risk

What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

Cause

How does it happen

Buffer Overflows can manifest in numerous different variations. In it's most basic form, the attack controls a buffer, which is then copied to a smaller buffer without size verification. Because the attacker's source buffer is larger than the program's target buffer, the attacker's data overwrites whatever is next on the stack, allowing the attacker to control program structures.
Alternatively, the vulnerability could be the result of improper bounds checking; exposing internal memory addresses outside of their valid scope; allowing the attacker to control the size of the target buffer; or various other forms.

General Recommendations

How to avoid it

- Always perform proper bounds checking before copying buffers or strings.
- Prefer to use safer functions and structures, e.g. safe string classes over `char*`, `strncpy` over `strcpy`, and so on.
- Consistently apply tests for the size of buffers.
- Do not return variable addresses outside the scope of their variables.

Source Code Examples

MemoryFree on StackVariable

Risk

What might happen

Undefined Behavior may result with a crash. Crashes may give an attacker valuable information about the system and the program internals. Furthermore, it may leave unprotected files (e.g memory) that may be exploited.

Cause

How does it happen

Calling free() on a variable that was not dynamically allocated (e.g. malloc) will result with an Undefined Behavior.

General Recommendations

How to avoid it

Use free() only on dynamically allocated variables in order to prevent unexpected behavior from the compiler.

Source Code Examples

CPP

Bad - Calling free() on a static variable

```
void clean_up(){
  char temp[256];
  do_something();
  free(tmp);
  return;
}
```

Good - Calling free() only on variables that were dynamically allocated

```
void clean_up(){
  char *buff;
  buff = (char*) malloc(1024);
  free(buff);
  return;
}
```

Wrong Size t Allocation

Risk

What might happen

Incorrect allocation of memory may result in unexpected behavior by either overwriting sections of memory with unexpected values. Under certain conditions where both an incorrect allocation of memory and the values being written can be controlled by an attacker, such an issue may result in execution of malicious code.

Cause

How does it happen

Some memory allocation functions require a size value to be provided as a parameter. The allocated size should be derived from the provided value, by providing the length value of the intended source, multiplied by the size of that length. Failure to perform the correct arithmetic to obtain the exact size of the value will likely result in the source overflowing its destination.

General Recommendations

How to avoid it

- Always perform the correct arithmetic to determine size.
- Specifically for memory allocation, calculate the allocation size from the allocation source:
    - Derive the size value from the length of intended source to determine the amount of units to be processed.
    - Always programmatically consider the size of the each unit and their conversion to memory units - for example, by using sizeof() on the unit's type.
    - Memory allocation should be a multiplication of the amount of units being written, times the size of each unit.

Source Code Examples

CPP

Allocating and Assigning Memory without Sizeof Arithmetic

```
int *ptr;
ptr = (int*)malloc(5);
for (int i = 0; i < 5; i++)
{
      ptr[i] = i * 2 + 1;
}
```

Allocating and Assigning Memory with Sizeof Arithmetic

```
int *ptr;
ptr = (int*)malloc(5 * sizeof(int));
for (int i = 0; i < 5; i++)
{
      ptr[i] = i * 2 + 1;
```

```
}
```

Incorrect Arithmetic of Multi-Byte String Allocation

```
wchar_t * dest;
dest = (wchar_t *)malloc(wcslen(source) + 1); // Would not crash for a short "source"
wcscpy((wchar_t *)dest, source);
wprintf(L"Dest: %s\r\n", dest);
```

Correct Arithmetic of Multi-Byte String Allocation

```
wchar_t * dest;
dest = (wchar_t *)malloc((wcslen(source) + 1) * sizeof(wchar_t));
wcscpy((wchar_t *)dest, source);
wprintf(L"Dest: %s\r\n", dest);
```

Dangerous Functions

Risk

What might happen

Use of dangerous functions may expose varying risks associated with each particular function, with potential impact of improper usage of these functions varying significantly. The presence of such functions indicates a flaw in code maintenance policies and adherence to secure coding practices, in a way that has allowed introducing known dangerous code into the application.

Cause

How does it happen

A dangerous function has been identified within the code. Functions are often deemed dangerous to use for numerous reasons, as there are different sets of vulnerabilities associated with usage of such functions. For example, some string copy and concatenation functions are vulnerable to Buffer Overflow, Memory Disclosure, Denial of Service and more. Use of these functions is not recommended.

General Recommendations

How to avoid it

- Deploy a secure and recommended alternative to any functions that were identified as dangerous.
    - If no secure alternative is found, conduct further researching and testing to identify whether current usage successfully sanitizes and verifies values, and thus successfully avoids the use-cases for whom the function is indeed dangerous
- Conduct a periodical review of methods that are in use, to ensure that all external libraries and built-in functions are up-to-date and whose use has not been excluded from best secure coding practices.

Source Code Examples

CPP

Buffer Overflow in gets()

```
int main()
{
	char buf[10];

	printf("Please enter your name: ");
	gets(buf); // veryveryverylongname
	if (buf == ACCEPTED_NAME)
	{
		// Do something
	}
	return 0;
}
```

Safe reading from user

```
int main()

{

	char buf[10];

	printf("Please enter your name: ");
	fgets(buf, sizeof(buf), stdin); //setting the amount of bytes to read
	if (buf == ACCEPTED_NAME)
	{
		//Do something
	}
	return 0;
}
```

Unsafe function for string copy

```
int main(int argc, char* argv[])

{

	char buf[10];
	strcpy(buf, argv[1]); // overflow occurs when len(argv[1]) > 10 bytes

	return 0;
}
```

Safe string copy

```
int main(int argc, char* argv[])

{

	char buf[10];
	strncpy(buf, argv[1], sizeof(buf));
	buf[9]= '\0'; //strncpy doesn't NULL terminates

	return 0;
}
```

Unsafe format string

```
int main(int argc, char* argv[])

{

	printf(argv[1]); // If argv[1] contains a format token, such as %s,%x or %d, will cause
an access violation
	return 0;
}
```

Safe format string

```
int main(int argc, char* argv[])

{

	printf("%s", argv[1]); // Second parameter is not a formattable string

	return 0;
}
```

Double Free

Weakness ID: 415 *(Weakness Variant)* **Status:** Draft

Description

Description Summary

The product calls free() twice on the same memory address, potentially leading to modification of unexpected memory locations.

Extended Description

When a program calls free() twice with the same argument, the program's memory management data structures become corrupted. This corruption can cause the program to crash or, in some circumstances, cause two later calls to malloc() to return the same pointer. If malloc() returns the same value twice and the program later gives the attacker control over the data that is written into this doubly-allocated memory, the program becomes vulnerable to a buffer overflow attack.

Alternate Terms

Double-free

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Languages

C

C++

Common Consequences

Scope	Effect
Access Control	Doubly freeing memory may result in a write-what-where condition, allowing an attacker to execute arbitrary code.

Likelihood of Exploit

Low to Medium

Demonstrative Examples

Example 1

The following code shows a simple example of a double free vulnerability.

(Bad Code)

Example Language: **C**

```
char* ptr = (char*)malloc (SIZE);
...
if (abrt) {
free(ptr);
}
...
free(ptr);
```

Double free vulnerabilities have two common (and sometimes overlapping) causes:

- Error conditions and other exceptional circumstances
- Confusion over which part of the program is responsible for freeing the memory

Although some double free vulnerabilities are not much more complicated than the previous example, most are spread out across hundreds of lines of code or even different files. Programmers seem particularly susceptible to freeing global variables

more than once.

Example 2

While contrived, this code should be exploitable on Linux distributions which do not ship with heap-chunk check summing turned on.

(Bad Code)

Example Language: **C**

```
#include <stdio.h>
#include <unistd.h>
#define BUFSIZE1 512
#define BUFSIZE2 ((BUFSIZE1/2) - 8)

int main(int argc, char **argv) {
char *buf1R1;
char *buf2R1;
char *buf1R2;
buf1R1 = (char *) malloc(BUFSIZE2);
buf2R1 = (char *) malloc(BUFSIZE2);
free(buf1R1);
free(buf2R1);
buf1R2 = (char *) malloc(BUFSIZE1);
strncpy(buf1R2, argv[1], BUFSIZE1-1);
free(buf2R1);
free(buf1R2);
}
```

Observed Examples

Reference	Description
CVE-2004-0642	Double free resultant from certain error conditions.
CVE-2004-0772	Double free resultant from certain error conditions.
CVE-2005-1689	Double free resultant from certain error conditions.
CVE-2003-0545	Double free from invalid ASN.1 encoding.
CVE-2003-1048	Double free from malformed GIF.
CVE-2005-0891	Double free from malformed GIF.
CVE-2002-0059	Double free from malformed compressed data.

Potential Mitigations

Phase: Architecture and Design

Choose a language that provides automatic memory management.

Phase: Implementation

Ensure that each allocation is freed only once. After freeing a chunk, set the pointer to NULL to ensure the pointer cannot be freed again. In complicated error conditions, be sure that clean-up routines respect the state of allocation properly. If the language is object oriented, ensure that object destructors delete each chunk of memory only once.

Phase: Implementation

Use a static analysis tool to find double free instances.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	398	Indicator of Poor Code Quality	**Seven Pernicious Kingdoms (primary)700**
ChildOf	Category	399	Resource Management Errors	**Development Concepts (primary)699**
ChildOf	Category	633	Weaknesses that Affect Memory	**Resource-specific Weaknesses (primary)631**
ChildOf	Weakness Base	666	Operation on Resource in Wrong Phase of	**Research Concepts (primary)1000**

			Lifetime	
ChildOf	Weakness Class	675	Duplicate Operations on Resource	Research Concepts1000
ChildOf	Category	742	CERT C Secure Coding Section 08 - Memory Management (MEM)	**Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734**
PeerOf	Weakness Base	123	Write-what-where Condition	Research Concepts1000
PeerOf	Weakness Base	416	Use After Free	Development Concepts699 Research Concepts1000
MemberOf	View	630	Weaknesses Examined by SAMATE	**Weaknesses Examined by SAMATE (primary)630**
PeerOf	Weakness Base	364	Signal Handler Race Condition	Research Concepts1000

Relationship Notes

This is usually resultant from another weakness, such as an unhandled error or race condition between threads. It could also be primary to weaknesses such as buffer overflows.

Affected Resources

- Memory

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
PLOVER			DFREE - Double-Free Vulnerability
7 Pernicious Kingdoms			Double Free
CLASP			Doubly freeing memory
CERT C Secure Coding	MEM00-C		Allocate and free memory in the same module, at the same level of abstraction
CERT C Secure Coding	MEM01-C		Store a new value in pointers immediately after free()
CERT C Secure Coding	MEM31-C		Free dynamically allocated memory exactly once

White Box Definitions

A weakness where code path has:

1. start statement that relinquishes a dynamically allocated memory resource
2. end statement that relinquishes the dynamically allocated memory resource

Maintenance Notes

It could be argued that Double Free would be most appropriately located as a child of "Use after Free", but "Use" and "Release" are considered to be distinct operations within vulnerability theory, therefore this is more accurately "Release of a Resource after Expiration or Release", which doesn't exist yet.

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	PLOVER		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Potential Mitigations, Time of Introduction		
2008-08-01		KDM Analytics	External
	added/updated white box definitions		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Description, Maintenance Notes, Relationships, Other Notes, Relationship Notes, Taxonomy Mappings		
2008-11-24	CWE Content Team	MITRE	Internal

	updated Relationships, Taxonomy Mappings		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Other Notes		

BACK TO TOP

Improper Sanitization of Special Elements used in a Command ('Command Injection')

Weakness ID: 77 *(Weakness Class)* **Status:** Draft

Description

Description Summary

The software constructs all or part of a command using externally-influenced input from an upstream component, but it does not sanitize or incorrectly sanitizes special elements that could modify the intended command when it is sent to a downstream component.

Extended Description

Command injection vulnerabilities typically occur when:

1. Data enters the application from an untrusted source.

2. The data is part of a string that is executed as a command by the application.

3. By executing the command, the application gives an attacker a privilege or capability that the attacker would not otherwise have.

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Languages

All

Common Consequences

Scope	Effect
Access Control	Command injection allows for the execution of arbitrary commands and code by the attacker.
Integrity	If a malicious user injects a character (such as a semi-colon) that delimits the end of one command and the beginning of another, it may be possible to then insert an entirely new and unrelated command that was not intended to be executed.

Likelihood of Exploit

Very High

Demonstrative Examples

Example 1

The following simple program accepts a filename as a command line argument and displays the contents of the file back to the user. The program is installed setuid root because it is intended for use as a learning tool to allow system administrators in-training to inspect privileged system files without giving them the ability to modify them or damage the system.

Example Language: **C**

```
int main(char* argc, char** argv) {
char cmd[CMD_MAX] = "/usr/bin/cat ";
strcat(cmd, argv[1]);
system(cmd);
}
```

Because the program runs with root privileges, the call to system() also executes with root privileges. If a user specifies a standard filename, the call works as expected. However, if an attacker passes a string of the form ";rm -rf /", then the call to system() fails to execute cat due to a lack of arguments and then plows on to recursively delete

the contents of the root partition.

Example 2

The following code is from an administrative web application designed to allow users to kick off a backup of an Oracle database using a batch-file wrapper around the rman utility and then run a cleanup.bat script to delete some temporary files. The script rmanDB.bat accepts a single command line parameter, which specifies what type of backup to perform. Because access to the database is restricted, the application runs the backup as a privileged user.

(Bad Code)

Example Language: **Java**

```
...
String btype = request.getParameter("backuptype");
String cmd = new String("cmd.exe /K \"
c:\\util\\rmanDB.bat "
+btype+
"&&c:\\utl\\cleanup.bat\"")
System.Runtime.getRuntime().exec(cmd);
...
```

The problem here is that the program does not do any validation on the backuptype parameter read from the user. Typically the Runtime.exec() function will not execute multiple commands, but in this case the program first runs the cmd.exe shell in order to run multiple commands with a single call to Runtime.exec(). Once the shell is invoked, it will happily execute multiple commands separated by two ampersands. If an attacker passes a string of the form "& del c:\\dbms*.*", then the application will execute this command along with the others specified by the program. Because of the nature of the application, it runs with the privileges necessary to interact with the database, which means whatever command the attacker injects will run with those privileges as well.

Example 3

The following code from a system utility uses the system property APPHOME to determine the directory in which it is installed and then executes an initialization script based on a relative path from the specified directory.

(Bad Code)

Example Language: **Java**

```
...
String home = System.getProperty("APPHOME");
String cmd = home + INITCMD;
java.lang.Runtime.getRuntime().exec(cmd);
...
```

The code above allows an attacker to execute arbitrary commands with the elevated privilege of the application by modifying the system property APPHOME to point to a different path containing a malicious version of INITCMD. Because the program does not validate the value read from the environment, if an attacker can control the value of the system property APPHOME, then they can fool the application into running malicious code and take control of the system.

Example 4

The following code is from a web application that allows users access to an interface through which they can update their password on the system. Part of the process for updating passwords in certain network environments is to run a make command in the /var/yp directory, the code for which is shown below.

(Bad Code)

Example Language: **Java**

```
...
System.Runtime.getRuntime().exec("make");
...
```

The problem here is that the program does not specify an absolute path for make and fails to clean its environment prior to executing the call to Runtime.exec(). If an attacker can modify the $PATH variable to point to a malicious binary called make and cause the program to be executed in their environment, then the malicious binary will be loaded instead of the one intended. Because of the nature of the application, it runs with the privileges necessary to perform system operations, which means the attacker's make will now be run with these privileges, possibly giving the attacker complete control of the system.

Example 5

The following code is a wrapper around the UNIX command cat which prints the contents of a file to standard out. It is also injectable:

(Bad Code)

Example Language: **C**

```
#include <stdio.h>
#include <unistd.h>

int main(int argc, char **argv) {

char cat[] = "cat ";
char *command;
size_t commandLength;

commandLength = strlen(cat) + strlen(argv[1]) + 1;
command = (char *) malloc(commandLength);
strncpy(command, cat, commandLength);
strncat(command, argv[1], (commandLength - strlen(cat)) );

system(command);
return (0);
}
```

Used normally, the output is simply the contents of the file requested:

```
$ ./catWrapper Story.txt
When last we left our heroes...
```

However, if we add a semicolon and another command to the end of this line, the command is executed by catWrapper with no complaint:

(Attack)

```
$ ./catWrapper Story.txt; ls
When last we left our heroes...
Story.txt
SensitiveFile.txt
PrivateData.db
a.out*
```

If catWrapper had been set to have a higher privilege level than the standard user, arbitrary commands could be executed with that higher privilege.

Potential Mitigations

Phase: Architecture and Design

If at all possible, use library calls rather than external processes to recreate the desired functionality

Phase: Implementation

If possible, ensure that all external commands called from the program are statically created.

Phase: Implementation

Strategy: Input Validation

Assume all input is malicious. Use an "accept known good" input validation strategy, i.e., use a whitelist of acceptable inputs that strictly conform to specifications. Reject any input that does not strictly conform to specifications, or transform it into something that does. Do not rely exclusively on looking for malicious or malformed inputs (i.e., do not rely on a blacklist). However, blacklists

can be useful for detecting potential attacks or determining which inputs are so malformed that they should be rejected outright.

When performing input validation, consider all potentially relevant properties, including length, type of input, the full range of acceptable values, missing or extra inputs, syntax, consistency across related fields, and conformance to business rules. As an example of business rule logic, "boat" may be syntactically valid because it only contains alphanumeric characters, but it is not valid if you are expecting colors such as "red" or "blue."

Run time: Run time policy enforcement may be used in a white-list fashion to prevent use of any non-sanctioned commands.

Assign permissions to the software system that prevents the user from accessing/opening privileged files.

Other Notes

Command injection is a common problem with wrapper programs.

Weakness Ordinalities

Ordinality	Description
Primary	*(where the weakness exists independent of other weaknesses)*

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	20	Improper Input Validation	**Seven Pernicious Kingdoms (primary)700**
ChildOf	Weakness Class	74	Failure to Sanitize Data into a Different Plane ('Injection')	**Development Concepts (primary)699 Research Concepts (primary)1000**
ChildOf	Category	713	OWASP Top Ten 2007 Category A2 - Injection Flaws	**Weaknesses in OWASP Top Ten (2007) (primary)629**
ChildOf	Category	722	OWASP Top Ten 2004 Category A1 - Unvalidated Input	Weaknesses in OWASP Top Ten (2004)711
ChildOf	Category	727	OWASP Top Ten 2004 Category A6 - Injection Flaws	**Weaknesses in OWASP Top Ten (2004) (primary)711**
ParentOf	Weakness Base	78	Improper Sanitization of Special Elements used in an OS Command ('OS Command Injection')	**Development Concepts (primary)699 Research Concepts (primary)1000**
ParentOf	Weakness Base	88	Argument Injection or Modification	**Development Concepts (primary)699 Research Concepts (primary)1000**
ParentOf	Weakness Base	89	Improper Sanitization of Special Elements used in an SQL Command ('SQL Injection')	**Development Concepts (primary)699 Research Concepts (primary)1000**
ParentOf	Weakness Base	90	Failure to Sanitize Data into LDAP Queries ('LDAP Injection')	**Development Concepts (primary)699 Research Concepts (primary)1000**
ParentOf	Weakness Base	624	Executable Regular Expression Error	**Development Concepts (primary)699 Research Concepts (primary)1000**

f Causal Nature

Explicit

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
7 Pernicious Kingdoms			Command Injection
CLASP			Command injection

OWASP Top Ten 2007	A2	CWE More Specific	Injection Flaws
OWASP Top Ten 2004	A1	CWE More Specific	Unvalidated Input
OWASP Top Ten 2004	A6	CWE More Specific	Injection Flaws

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
15	Command Delimiters	
23	File System Function Injection, Content Based	
43	Exploiting Multiple Input Interpretation Layers	
75	Manipulating Writeable Configuration Files	
6	Argument Injection	
11	Cause Web Server Misclassification	
76	Manipulating Input to File System Calls	

References

G. Hoglund and G. McGraw. "Exploiting Software: How to Break Code". Addison-Wesley. February 2004.

Content History

Submissions

Submission Date	Submitter	Organization	Source
	7 Pernicious Kingdoms		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-15		Veracode	External
	Suggested OWASP Top Ten 2004 mapping		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Common Consequences, Relationships, Other Notes, Taxonomy Mappings, Weakness Ordinalities		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples, Name		
2009-07-27	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples, Description, Name		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Common Consequences, Description, Other Notes, Potential Mitigations		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Potential Mitigations, Relationships		

Previous Entry Names

Change Date	Previous Entry Name
2008-04-11	Command Injection
2009-05-27	Failure to Sanitize Data into a Control Plane (aka 'Command Injection')
2009-07-27	Failure to Sanitize Data into a Control Plane ('Command Injection')

BACK TO TOP

Failure to Release Memory Before Removing Last Reference ('Memory Leak')

Weakness ID: 401 *(Weakness Base)* **Status:** Draft

Description

Description Summary

The software does not sufficiently track and release allocated memory after it has been used, which slowly consumes remaining memory.

Extended Description

This is often triggered by improper handling of malformed data or unexpectedly interrupted sessions.

Terminology Notes

"memory leak" has sometimes been used to describe other kinds of issues, e.g. for information leaks in which the contents of memory are inadvertently leaked (CVE-2003-0400 is one such example of this terminology conflict).

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Languages

C

C++

Modes of Introduction

Memory leaks have two common and sometimes overlapping causes:

- Error conditions and other exceptional circumstances
- Confusion over which part of the program is responsible for freeing the memory

Common Consequences

Scope	Effect
Availability	Most memory leaks result in general software reliability problems, but if an attacker can intentionally trigger a memory leak, the attacker might be able to launch a denial of service attack (by crashing or hanging the program) or take advantage of other unexpected program behavior resulting from a low memory condition.

Likelihood of Exploit

Medium

Demonstrative Examples

Example 1

The following C function leaks a block of allocated memory if the call to read() fails to return the expected number of bytes:

(Bad Code)

Example Language: **C**

```
char* getBlock(int fd) {
char* buf = (char*) malloc(BLOCK_SIZE);
if (!buf) {
return NULL;
}
if (read(fd, buf, BLOCK_SIZE) != BLOCK_SIZE) {

return NULL;
}
```

```
return buf;
}
```

Example 2

Here the problem is that every time a connection is made, more memory is allocated. So if one just opened up more and more connections, eventually the machine would run out of memory.

(Bad Code)

Example Language: **C**

```
bar connection(){
foo = malloc(1024);
return foo;
}
endConnection(bar foo) {

free(foo);
}
int main() {

while(1) //thread 1
//On a connection
foo=connection(); //thread 2
//When the connection ends
endConnection(foo)
}
```

Observed Examples

Reference	Description
CVE-2005-3119	Memory leak because function does not free() an element of a data structure.
CVE-2004-0427	Memory leak when counter variable is not decremented.
CVE-2002-0574	Memory leak when counter variable is not decremented.
CVE-2005-3181	Kernel uses wrong function to release a data structure, preventing data from being properly tracked by other code.
CVE-2004-0222	Memory leak via unknown manipulations as part of protocol test suite.
CVE-2001-0136	Memory leak via a series of the same command.

Potential Mitigations

Pre-design: Use a language or compiler that performs automatic bounds checking.

Phase: Architecture and Design

Use an abstraction library to abstract away risky APIs. Not a complete solution.

Pre-design through Build: The Boehm-Demers-Weiser Garbage Collector or valgrind can be used to detect leaks in code. This is not a complete solution as it is not 100% effective.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	398	Indicator of Poor Code Quality	**Seven Pernicious Kingdoms (primary)700**
ChildOf	Category	399	Resource Management Errors	**Development Concepts (primary)699**
ChildOf	Category	633	Weaknesses that Affect Memory	**Resource-specific Weaknesses (primary)631**
ChildOf	Category	730	OWASP Top Ten 2004 Category A9 - Denial of Service	**Weaknesses in OWASP Top Ten (2004) (primary)711**
ChildOf	Weakness Base	772	Missing Release of Resource after Effective	**Research Concepts (primary)1000**

			Lifetime	
MemberOf	View	630	Weaknesses Examined by SAMATE	**Weaknesses Examined by SAMATE (primary)630**
CanFollow	Weakness Class	390	Detection of Error Condition Without Action	Research Concepts1000

Relationship Notes

This is often a resultant weakness due to improper handling of malformed data or early termination of sessions.

Affected Resources

- Memory

Functional Areas

- Memory management

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
PLOVER			Memory leak
7 Pernicious Kingdoms			Memory Leak
CLASP			Failure to deallocate data
OWASP Top Ten 2004	A9	CWE More Specific	Denial of Service

White Box Definitions

A weakness where the code path has:

1. start statement that allocates dynamically allocated memory resource

2. end statement that loses identity of the dynamically allocated memory resource creating situation where dynamically allocated memory resource is never relinquished

Where "loses" is defined through the following scenarios:

1. identity of the dynamic allocated memory resource never obtained

2. the statement assigns another value to the data element that stored the identity of the dynamically allocated memory resource and there are no aliases of that data element

3. identity of the dynamic allocated memory resource obtained but never passed on to function for memory resource release

4. the data element that stored the identity of the dynamically allocated resource has reached the end of its scope at the statement and there are no aliases of that data element

References

J. Whittaker and H. Thompson. "How to Break Software Security". Addison Wesley. 2003.

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	PLOVER		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-01		KDM Analytics	External
	added/updated white box definitions		
2008-08-15		Veracode	External
	Suggested OWASP Top Ten 2004 mapping		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Relationships, Other Notes, References, Relationship Notes, Taxonomy Mappings, Terminology Notes		
2008-10-14	CWE Content Team	MITRE	Internal
	updated Description		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Other Notes		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Name		
2009-07-17	KDM Analytics		External
	Improved the White Box Definition		

2009-07-27	CWE Content Team	MITRE	Internal
	updated White Box Definitions		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Modes of Introduction, Other Notes		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Relationships		

Previous Entry Names

Change Date	Previous Entry Name
2008-04-11	Memory Leak
2009-05-27	Failure to Release Memory Before Removing Last Reference (aka 'Memory Leak')

BACK TO TOP

Inadequate Encryption Strength

Risk

What might happen

Using weak or outdated cryptography does not provide sufficient protection for sensitive data. An attacker that gains access to the encrypted data would likely be able to break the encryption, using either cryptanalysis or brute force attacks. Thus, the attacker would be able to steal user passwords and other personal data. This could lead to user impersonation or identity theft.

Cause

How does it happen

The application uses a weak algorithm, that is considered obselete since it is relatively easy to break. These obselete algorithms are vulnerable to several different kinds of attacks, including brute force.

General Recommendations

How to avoid it

Generic Guidance:

- Always use strong, modern algorithms for encryption, hashing, and so on.
- Do not use weak, outdated, or obsolete algorithms.
- Ensure you select the correct cryptographic mechanism according to the specific requirements.
- Passwords should be protected with a dedicated password protection scheme, such as bcrypt, scrypt, PBKDF2, or Argon2.

Specific Recommendations:

- Do not use SHA-1, MD5, or any other weak hash algorithm to protect passwords or personal data. Instead, use a stronger hash such as SHA-256 when a secure hash is required.
- Do not use DES, Triple-DES, RC2, or any other weak encryption algorithm to protect passwords or personal data. Instead, use a stronger encryption algorithm such as AES to protect personal data.
- Do not use weak encryption modes such as ECB, or rely on insecure defaults. Explicitly specify a stronger encryption mode, such as GCM.
- For symmetric encryption, use a key length of at least 256 bits.

Source Code Examples

Java

Weakly Hashed PII

```
string protectSSN(HttpServletRequest req) {
    string socialSecurityNum = req.getParameter("SocialSecurityNo");

    return DigestUtils.md5Hex(socialSecurityNum);
}
```

Stronger Hash for PII

```
string protectSSN(HttpServletRequest req) {
    string socialSecurityNum = req.getParameter("SocialSecurityNo");

    return DigestUtils.sha256Hex(socialSecurityNum);
}
```

Use After Free

Risk

What might happen

A use after free error will cause code to use an area of memory previously assigned with a specific value, which has since been freed and may have been overwritten by another value. This error will likely cause unexpected behavior, memory corruption and crash errors. In some cases where the freed and used section of memory is used to determine execution flow, and the error can be induced by an attacker, this may result in execution of malicious code.

Cause

How does it happen

Pointers to variables allow code to have an address with a set size to a dynamically allocated variable. Eventually, the pointer's destination may become free - either explicitly in code, such as when programmatically freeing this variable, or implicitly, such as when a local variable is returned - once it is returned, the variable's scope is released. Once freed, this memory will be re-used by the application, overwritten with new data. At this point, dereferencing this pointer will potentially resolve newly written and unexpected data.

General Recommendations

How to avoid it

- Do not return local variables or pointers
- Review code to ensure no flow allows use of a pointer after it has been explicitly freed

---

Source Code Examples

CPP

Use of Variable after It was Freed

```
free(input);
printf("%s", input);
```

Use of Pointer to Local Variable That Was Freed On Return

```
int* func1()
{
	int i;
	i = 1;
	return &i;
}

void func2()

{

	int j;
	j = 5;
```

```
}

//..
	int * i = func1();
	printf("%d\r\n", *i); // Output could be 1 or Segmentation Fault
	func2();
	printf("%d\r\n", *i); // Output is 5, which is j's value, as func2() overwrote data in
the stack
//..
```

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());
```

Stored Buffer Overflow fgets

Risk

What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

Cause

How does it happen

Buffer Overflows can manifest in numerous different variations. In it's most basic form, the attack controls a buffer, which is then copied to a smaller buffer without size verification. Because the attacker's source buffer is larger than the program's target buffer, the attacker's data overwrites whatever is next on the stack, allowing the attacker to control program structures.
Alternatively, the vulnerability could be the result of improper bounds checking; exposing internal memory addresses outside of their valid scope; allowing the attacker to control the size of the target buffer; or various other forms.

General Recommendations

How to avoid it

- Always perform proper bounds checking before copying buffers or strings.
- Prefer to use safer functions and structures, e.g. safe string classes over `char*`, `strncpy` over `strcpy`, and so on.
- Consistently apply tests for the size of buffers.
- Do not return variable addresses outside the scope of their variables.

Source Code Examples

CPP

Overflowing Buffers

```
const int BUFFER_SIZE = 10;
char buffer[BUFFER_SIZE];

void copyStringToBuffer(char* inputString)
{
    strcpy(buffer, inputString);
}
```

Checked Buffers

```
const int BUFFER_SIZE = 10;
const int MAX_INPUT_SIZE = 256;
char buffer[BUFFER_SIZE];

void copyStringToBuffer(char* inputString)
```

```
{
    if (strnlen(inputString, MAX_INPUT_SIZE) < sizeof(buffer))
    {
        strncpy(buffer, inputString, sizeof(buffer));
    }
}
```

Potential Off by One Error in Loops

Risk

What might happen

An off by one error may result in overwriting or over-reading of unintended memory; in most cases, this can result in unexpected behavior and even application crashes. In other cases, where allocation can be controlled by an attacker, a combination of variable assignment and an off by one error can result in execution of malicious code.

Cause

How does it happen

Often when designating variables to memory, a calculation error may occur when determining size or length that is off by one.

For example in loops, when allocating an array of size 2, its cells are counted as 0,1 - therefore, if a For loop iterator on the array is incorrectly set with the start condition i=0 and the continuation condition i<=2, three cells will be accessed instead of 2, and an attempt will be made to write or read cell [2], which was not originally allocated, resulting in potential corruption of memory outside the bounds of the originally assigned array.

Another example occurs when a null-byte terminated string, in the form of a character array, is copied without its terminating null-byte. Without the null-byte, the string representation is unterminated, resulting in certain functions to over-read memory as they expect the missing null terminator.

General Recommendations

How to avoid it

- Always ensure that a given iteration boundary is correct:
  - With array iterations, consider that arrays begin with cell 0 and end with cell n-1, for a size n array.
  - With character arrays and null-byte terminated string representations, consider that the null byte is required and should not be overwritten or ignored; ensure functions in use are not vulnerable to off-by-one, specifically for instances where null-bytes are automatically appended after the buffer, instead of in place of its last character.
- Where possible, use safe functions that manage memory and are not prone to off-by-one errors.

Source Code Examples

CPP

Off-By-One in For Loop

```
int *ptr;
ptr = (int*)malloc(5 * sizeof(int));
for (int i = 0; i <= 5; i++)
{
      ptr[i] = i * 2 + 1; // ptr[5] will be set, but is out of bounds
```

```
}
```

Proper Iteration in For Loop

```
int *ptr;
ptr = (int*)malloc(5 * sizeof(int));
for (int i = 0; i < 5; i++)
{
      ptr[i] = i * 2 + 1; // ptr[0-4] are well defined
}
```

Off-By-One in strncat

```
strncat(buf, input, sizeof(buf) - strlen(buf)); // actual value should be sizeof(buf)-
strlen(buf)-1 - this form will overwrite the terminating nullbyte
```

Heuristic Buffer Overflow malloc

Risk

What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

Cause

How does it happen

Buffer Overflows can manifest in numerous different variations. In it's most basic form, the attack controls a buffer, which is then copied to a smaller buffer without size verification. Because the attacker's source buffer is larger than the program's target buffer, the attacker's data overwrites whatever is next on the stack, allowing the attacker to control program structures.
Alternatively, the vulnerability could be the result of improper bounds checking; exposing internal memory addresses outside of their valid scope; allowing the attacker to control the size of the target buffer; or various other forms.

General Recommendations

How to avoid it

- Always perform proper bounds checking before copying buffers or strings.
- Prefer to use safer functions and structures, e.g. safe string classes over `char*`, `strncpy` over `strcpy`, and so on.
- Consistently apply tests for the size of buffers.
- Do not return variable addresses outside the scope of their variables.

Source Code Examples

Indicator of Poor Code Quality

Weakness ID: 398 *(Weakness Class)* **Status:** Draft

Description

Description Summary

The code has features that do not directly introduce a weakness or vulnerability, but indicate that the product has not been carefully developed or maintained.

Extended Description

Programs are more likely to be secure when good development practices are followed. If a program is complex, difficult to maintain, not portable, or shows evidence of neglect, then there is a higher likelihood that weaknesses are buried in the code.

Time of Introduction

- Architecture and Design
- Implementation

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	18	Source Code	**Development Concepts (primary)699**
ChildOf	Weakness Class	710	Coding Standards Violation	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	107	Struts: Unused Validation Form	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	110	Struts: Validator Without Form Field	**Research Concepts (primary)1000**
ParentOf	Category	399	Resource Management Errors	**Development Concepts (primary)699**
ParentOf	Weakness Base	401	Failure to Release Memory Before Removing Last Reference ('Memory Leak')	**Seven Pernicious Kingdoms (primary)700**
ParentOf	Weakness Base	404	Improper Resource Shutdown or Release	Development Concepts699 **Seven Pernicious Kingdoms (primary)700**
ParentOf	Weakness Variant	415	Double Free	**Seven Pernicious Kingdoms (primary)700**
ParentOf	Weakness Base	416	Use After Free	**Seven Pernicious Kingdoms (primary)700**
ParentOf	Weakness Variant	457	Use of Uninitialized Variable	**Seven Pernicious Kingdoms (primary)700**
ParentOf	Weakness Base	474	Use of Function with Inconsistent Implementations	**Development Concepts (primary)699 Seven Pernicious Kingdoms (primary)700 Research Concepts (primary)1000**
ParentOf	Weakness Base	475	Undefined Behavior for Input to API	**Development Concepts (primary)699 Seven Pernicious Kingdoms (primary)700**
ParentOf	Weakness Base	476	NULL Pointer	**Development**

			Dereference	**Concepts (primary)699 Seven Pernicious Kingdoms (primary)700 Research Concepts (primary)1000**
ParentOf	Weakness Base	477	Use of Obsolete Functions	**Development Concepts (primary)699 Seven Pernicious Kingdoms (primary)700 Research Concepts (primary)1000**
ParentOf	Weakness Variant	478	Missing Default Case in Switch Statement	**Development Concepts (primary)699**
ParentOf	Weakness Variant	479	Unsafe Function Call from a Signal Handler	**Development Concepts (primary)699**
ParentOf	Weakness Variant	483	Incorrect Block Delimitation	**Development Concepts (primary)699**
ParentOf	Weakness Base	484	Omitted Break Statement in Switch	**Development Concepts (primary)699** Research Concepts1000
ParentOf	Weakness Variant	546	Suspicious Comment	**Development Concepts (primary)699 Research Concepts (primary)1000**
ParentOf	Weakness Variant	547	Use of Hard-coded, Security-relevant Constants	**Development Concepts (primary)699 Research Concepts (primary)1000**
ParentOf	Weakness Variant	561	Dead Code	**Development Concepts (primary)699 Research Concepts (primary)1000**
ParentOf	Weakness Base	562	Return of Stack Variable Address	**Development Concepts (primary)699** Research Concepts1000
ParentOf	Weakness Variant	563	Unused Variable	**Development Concepts (primary)699 Research Concepts (primary)1000**
ParentOf	Category	569	Expression Issues	**Development Concepts (primary)699**
ParentOf	Weakness Variant	585	Empty Synchronized Block	**Development Concepts (primary)699 Research Concepts (primary)1000**
ParentOf	Weakness Variant	586	Explicit Call to Finalize()	**Development Concepts (primary)699**
ParentOf	Weakness Variant	617	Reachable Assertion	**Development Concepts (primary)699**
ParentOf	Weakness Base	676	Use of Potentially Dangerous Function	**Development Concepts (primary)699 Research Concepts (primary)1000**
MemberOf	View	700	Seven Pernicious Kingdoms	**Seven Pernicious Kingdoms (primary)700**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name

7 Pernicious Kingdoms			Code Quality

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	7 Pernicious Kingdoms		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Description, Relationships, Taxonomy Mappings		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Relationships		
Previous Entry Names			
Change Date	**Previous Entry Name**		
2008-04-11	Code Quality		

BACK TO TOP

Improper Access Control (Authorization)

Weakness ID: 285 *(Weakness Class)* **Status:** Draft

Description

Description Summary

The software does not perform or incorrectly performs access control checks across all potential execution paths.

Extended Description

When access control checks are not applied consistently - or not at all - users are able to access data or perform actions that they should not be allowed to perform. This can lead to a wide range of problems, including information leaks, denial of service, and arbitrary code execution.

Alternate Terms

AuthZ:	"AuthZ" is typically used as an abbreviation of "authorization" within the web application security community. It is also distinct from "AuthC," which is an abbreviation of "authentication." The use of "Auth" as an abbreviation is discouraged, since it could be used for either authentication or authorization.

Time of Introduction

- Architecture and Design
- Implementation
- Operation

Applicable Platforms

Languages

Language-independent

Technology Classes

Web-Server: *(Often)*

Database-Server: *(Often)*

Modes of Introduction

A developer may introduce authorization weaknesses because of a lack of understanding about the underlying technologies. For example, a developer may assume that attackers cannot modify certain inputs such as headers or cookies.

Authorization weaknesses may arise when a single-user application is ported to a multi-user environment.

Common Consequences

Scope	Effect
Confidentiality	An attacker could read sensitive data, either by reading the data directly from a data store that is not properly restricted, or by accessing insufficiently-protected, privileged functionality to read the data.
Integrity	An attacker could modify sensitive data, either by writing the data directly to a data store that is not properly restricted, or by accessing insufficiently-protected, privileged functionality to write the data.
Integrity	An attacker could gain privileges by modifying or reading critical data directly, or by accessing insufficiently-protected, privileged functionality.

Likelihood of Exploit

High

Detection Methods

Automated Static Analysis

Automated static analysis is useful for detecting commonly-used idioms for authorization. A tool may be able to analyze related configuration files, such as .htaccess in Apache web servers, or detect the usage of commonly-used authorization libraries.

Generally, automated static analysis tools have difficulty detecting custom authorization schemes. In addition, the software's design may include some functionality that is accessible to any user and does not require an authorization check; an automated technique that detects the absence of authorization may report false positives.

Effectiveness: Limited

Automated Dynamic Analysis

Automated dynamic analysis may find many or all possible interfaces that do not require authorization, but manual analysis is required to determine if the lack of authorization violates business logic

Manual Analysis

This weakness can be detected using tools and techniques that require manual (human) analysis, such as penetration testing, threat modeling, and interactive tools that allow the tester to record and modify an active session.

Specifically, manual static analysis is useful for evaluating the correctness of custom authorization mechanisms.

Effectiveness: Moderate

These may be more effective than strictly automated techniques. This is especially the case with weaknesses that are related to design and business rules. However, manual efforts might not achieve desired code coverage within limited time constraints.

Demonstrative Examples

Example 1

The following program could be part of a bulletin board system that allows users to send private messages to each other. This program intends to authenticate the user before deciding whether a private message should be displayed. Assume that LookupMessageObject() ensures that the $id argument is numeric, constructs a filename based on that id, and reads the message details from that file. Also assume that the program stores all private messages for all users in the same directory.

(Bad Code)

Example Language: **Perl**

```
sub DisplayPrivateMessage {
my($id) = @_;
my $Message = LookupMessageObject($id);
print "From: " . encodeHTML($Message->{from}) . "<br>\n";
print "Subject: " . encodeHTML($Message->{subject}) . "\n";
print "<hr>\n";
print "Body: " . encodeHTML($Message->{body}) . "\n";
}

my $q = new CGI;
# For purposes of this example, assume that CWE-309 and
# CWE-523 do not apply.
if (! AuthenticateUser($q->param('username'), $q->param('password'))) {
ExitError("invalid username or password");
}

my $id = $q->param('id');
DisplayPrivateMessage($id);
```

While the program properly exits if authentication fails, it does not ensure that the message is addressed to the user. As a result, an authenticated attacker could provide any arbitrary identifier and read private messages that were intended for other users.

One way to avoid this problem would be to ensure that the "to" field in the message object matches the username of the authenticated user.

Observed Examples

Reference	Description
CVE-2009-3168	Web application does not restrict access to admin scripts, allowing authenticated users to reset administrative passwords.

CVE-2009-2960	Web application does not restrict access to admin scripts, allowing authenticated users to modify passwords of other users.
CVE-2009-3597	Web application stores database file under the web root with insufficient access control (CWE-219), allowing direct request.
CVE-2009-2282	Terminal server does not check authorization for guest access.
CVE-2009-3230	Database server does not use appropriate privileges for certain sensitive operations.
CVE-2009-2213	Gateway uses default "Allow" configuration for its authorization settings.
CVE-2009-0034	Chain: product does not properly interpret a configuration option for a system group, allowing users to gain privileges.
CVE-2008-6123	Chain: SNMP product does not properly parse a configuration option for which hosts are allowed to connect, allowing unauthorized IP addresses to connect.
CVE-2008-5027	System monitoring software allows users to bypass authorization by creating custom forms.
CVE-2008-7109	Chain: reliance on client-side security (CWE-602) allows attackers to bypass authorization using a custom client.
CVE-2008-3424	Chain: product does not properly handle wildcards in an authorization policy list, allowing unintended access.
CVE-2009-3781	Content management system does not check access permissions for private files, allowing others to view those files.
CVE-2008-4577	ACL-based protection mechanism treats negative access rights as if they are positive, allowing bypass of intended restrictions.
CVE-2008-6548	Product does not check the ACL of a page accessed using an "include" directive, allowing attackers to read unauthorized files.
CVE-2007-2925	Default ACL list for a DNS server does not set certain ACLs, allowing unauthorized DNS queries.
CVE-2006-6679	Product relies on the X-Forwarded-For HTTP header for authorization, allowing unintended access by spoofing the header.
CVE-2005-3623	OS kernel does not check for a certain privilege before setting ACLs for files.
CVE-2005-2801	Chain: file-system code performs an incorrect comparison (CWE-697), preventing defauls ACLs from being properly applied.
CVE-2001-1155	Chain: product does not properly check the result of a reverse DNS lookup because of operator precedence (CWE-783), allowing bypass of DNS-based access restrictions.

Potential Mitigations

Phase: Architecture and Design

Divide your application into anonymous, normal, privileged, and administrative areas. Reduce the attack surface by carefully mapping roles with data and functionality. Use role-based access control (RBAC) to enforce the roles at the appropriate boundaries.

Note that this approach may not protect against horizontal authorization, i.e., it will not protect a user from attacking others with the same role.

Phase: Architecture and Design

Ensure that you perform access control checks related to your business logic. These checks may be different than the access control checks that you apply to more generic resources such as files, connections, processes, memory, and database records. For example, a database may restrict access for medical records to a specific database user, but each record might only be intended to be accessible to the patient and the patient's doctor.

Phase: Architecture and Design

Strategy: Libraries or Frameworks

Use a vetted library or framework that does not allow this weakness to occur or provides constructs that make this weakness

easier to avoid.

For example, consider using authorization frameworks such as the JAAS Authorization Framework and the OWASP ESAPI Access Control feature.

Phase: Architecture and Design

For web applications, make sure that the access control mechanism is enforced correctly at the server side on every page. Users should not be able to access any unauthorized functionality or information by simply requesting direct access to that page.

One way to do this is to ensure that all pages containing sensitive information are not cached, and that all such pages restrict access to requests that are accompanied by an active and authenticated session token associated with a user who has the required permissions to access that page.

Phases: System Configuration; Installation

Use the access control capabilities of your operating system and server environment and define your access control lists accordingly. Use a "default deny" policy when defining these ACLs.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	254	Security Features	**Seven Pernicious Kingdoms (primary)700**
ChildOf	Weakness Class	284	Access Control (Authorization) Issues	**Development Concepts (primary)699 Research Concepts (primary)1000**
ChildOf	Category	721	OWASP Top Ten 2007 Category A10 - Failure to Restrict URL Access	**Weaknesses in OWASP Top Ten (2007) (primary)629**
ChildOf	Category	723	OWASP Top Ten 2004 Category A2 - Broken Access Control	**Weaknesses in OWASP Top Ten (2004) (primary)711**
ChildOf	Category	753	2009 Top 25 - Porous Defenses	**Weaknesses in the 2009 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)750**
ChildOf	Category	803	2010 Top 25 - Porous Defenses	**Weaknesses in the 2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800**
ParentOf	Weakness Variant	219	Sensitive Data Under Web Root	**Research Concepts (primary)1000**
ParentOf	Weakness Base	551	Incorrect Behavior Order: Authorization Before Parsing and Canonicalization	**Development Concepts (primary)699** Research Concepts1000
ParentOf	Weakness Class	638	Failure to Use Complete Mediation	Research Concepts1000
ParentOf	Weakness Base	804	Guessable CAPTCHA	**Development Concepts (primary)699 Research Concepts (primary)1000**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
7 Pernicious Kingdoms			Missing Access Control
OWASP Top Ten 2007	A10	CWE More Specific	Failure to Restrict URL Access
OWASP Top Ten 2004	A2	CWE More Specific	Broken Access Control

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
1	Accessing Functionality Not Properly Constrained by ACLs	
13	Subverting Environment Variable Values	

17	Accessing, Modifying or Executing Executable Files
87	Forceful Browsing
39	Manipulating Opaque Client-based Data Tokens
45	Buffer Overflow via Symbolic Links
51	Poison Web Service Registry
59	Session Credential Falsification through Prediction
60	Reusing Session IDs (aka Session Replay)
77	Manipulating User-Controlled Variables
76	Manipulating Input to File System Calls
104	Cross Zone Scripting

References

NIST. "Role Based Access Control and Role Based Security". <http://csrc.nist.gov/groups/SNS/rbac/>.

[REF-11] M. Howard and D. LeBlanc. "Writing Secure Code". Chapter 4, "Authorization" Page 114; Chapter 6, "Determining Appropriate Access Control" Page 171. 2nd Edition. Microsoft. 2002.

Content History

Submissions

Submission Date	Submitter	Organization	Source
	7 Pernicious Kingdoms		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-15		Veracode	External
	Suggested OWASP Top Ten 2004 mapping		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Relationships, Other Notes, Taxonomy Mappings		
2009-01-12	CWE Content Team	MITRE	Internal
	updated Common Consequences, Description, Likelihood of Exploit, Name, Other Notes, Potential Mitigations, References, Relationships		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Potential Mitigations		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Description, Related Attack Patterns		
2009-07-27	CWE Content Team	MITRE	Internal
	updated Relationships		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Type		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Demonstrative Examples, Detection Factors, Modes of Introduction, Observed Examples, Relationships		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Alternate Terms, Detection Factors, Potential Mitigations, References, Relationships		
2010-04-05	CWE Content Team	MITRE	Internal
	updated Potential Mitigations		

Previous Entry Names

Change Date	Previous Entry Name
2009-01-12	Missing or Inconsistent Access Control

BACK TO TOP

Incorrect Permission Assignment for Critical Resource

Weakness ID: 732 *(Weakness Class)* **Status:** Draft

Description

Description Summary

The software specifies permissions for a security-critical resource in a way that allows that resource to be read or modified by unintended actors.

Extended Description

When a resource is given a permissions setting that provides access to a wider range of actors than required, it could lead to the disclosure of sensitive information, or the modification of that resource by unintended parties. This is especially dangerous when the resource is related to program configuration, execution or sensitive user data.

Time of Introduction

- Architecture and Design
- Implementation
- Installation
- Operation

Applicable Platforms

Languages

Language-independent

Modes of Introduction

The developer may set loose permissions in order to minimize problems when the user first runs the program, then create documentation stating that permissions should be tightened. Since system administrators and users do not always read the documentation, this can result in insecure permissions being left unchanged.

The developer might make certain assumptions about the environment in which the software runs - e.g., that the software is running on a single-user system, or the software is only accessible to trusted administrators. When the software is running in a different environment, the permissions become a problem.

Common Consequences

Scope	Effect
Confidentiality	An attacker may be able to read sensitive information from the associated resource, such as credentials or configuration information stored in a file.
Integrity	An attacker may be able to modify critical properties of the associated resource to gain privileges, such as replacing a world-writable executable with a Trojan horse.
Availability	An attacker may be able to destroy or corrupt critical data in the associated resource, such as deletion of records from a database.

Likelihood of Exploit

Medium to High

Detection Methods

Automated Static Analysis

Automated static analysis may be effective in detecting permission problems for system resources such as files, directories, shared memory, device interfaces, etc. Automated techniques may be able to detect the use of library functions that modify permissions, then analyze function calls for arguments that contain potentially insecure values.

However, since the software's intended security policy might allow loose permissions for certain operations (such as publishing a file on a web server), automated static analysis may produce some false positives - i.e., warnings that do not have any security consequences or require any code changes.

When custom permissions models are used - such as defining who can read messages in a particular forum in a bulletin board system - these can be difficult to detect using automated static analysis. It may be possible to define custom signatures that

identify any custom functions that implement the permission checks and assignments.

Automated Dynamic Analysis

Automated dynamic analysis may be effective in detecting permission problems for system resources such as files, directories, shared memory, device interfaces, etc.

However, since the software's intended security policy might allow loose permissions for certain operations (such as publishing a file on a web server), automated dynamic analysis may produce some false positives - i.e., warnings that do not have any security consequences or require any code changes.

When custom permissions models are used - such as defining who can read messages in a particular forum in a bulletin board system - these can be difficult to detect using automated dynamic analysis. It may be possible to define custom signatures that identify any custom functions that implement the permission checks and assignments.

Manual Static Analysis

Manual static analysis may be effective in detecting the use of custom permissions models and functions. The code could then be examined to identifying usage of the related functions. Then the human analyst could evaluate permission assignments in the context of the intended security model of the software.

Manual Dynamic Analysis

Manual dynamic analysis may be effective in detecting the use of custom permissions models and functions. The program could then be executed with a focus on exercising code paths that are related to the custom permissions. Then the human analyst could evaluate permission assignments in the context of the intended security model of the software.

Fuzzing

Fuzzing is not effective in detecting this weakness.

Demonstrative Examples

Example 1

The following code sets the umask of the process to 0 before creating a file and writing "Hello world" into the file.

(Bad Code)

Example Language: **C**

```
#define OUTFILE "hello.out"

umask(0);
FILE *out;
/* Ignore CWE-59 (link following) for brevity */
out = fopen(OUTFILE, "w");
if (out) {
fprintf(out, "hello world!\n");
fclose(out);
}
```

After running this program on a UNIX system, running the "ls -l" command might return the following output:

(Result)

```
-rw-rw-rw- 1 username 13 Nov 24 17:58 hello.out
```

The "rw-rw-rw-" string indicates that the owner, group, and world (all users) can read the file and write to it.

Example 2

The following code snippet might be used as a monitor to periodically record whether a web site is alive. To ensure that the file can always be modified, the code uses chmod() to make the file world-writable.

(Bad Code)

Example Language: **Perl**

```
$fileName = "secretFile.out";

if (-e $fileName) {
chmod 0777, $fileName;
}
```

```
my $outFH;
if (! open($outFH, ">>$fileName")) {
ExitError("Couldn't append to $fileName: $!");
}
my $dateString = FormatCurrentTime();
my $status = IsHostAlive("cwe.mitre.org");
print $outFH "$dateString cwe status: $status!\n";
close($outFH);
```

The first time the program runs, it might create a new file that inherits the permissions from its environment. A file listing might look like:

(Result)

```
-rw-r--r-- 1 username 13 Nov 24 17:58 secretFile.out
```

This listing might occur when the user has a default umask of 022, which is a common setting. Depending on the nature of the file, the user might not have intended to make it readable by everyone on the system.

The next time the program runs, however - and all subsequent executions - the chmod will set the file's permissions so that the owner, group, and world (all users) can read the file and write to it:

(Result)

```
-rw-rw-rw- 1 username 13 Nov 24 17:58 secretFile.out
```

Perhaps the programmer tried to do this because a different process uses different permissions that might prevent the file from being updated.

Example 3

The following command recursively sets world-readable permissions for a directory and all of its children:

(Bad Code)

Example Language: **Shell**

```
chmod -R ugo+r DIRNAME
```

If this command is run from a program, the person calling the program might not expect that all the files under the directory will be world-readable. If the directory is expected to contain private data, this could become a security problem.

Observed Examples

Reference	Description
CVE-2009-3482	Anti-virus product sets insecure "Everyone: Full Control" permissions for files under the "Program Files" folder, allowing attackers to replace executables with Trojan horses.
CVE-2009-3897	Product creates directories with 0777 permissions at installation, allowing users to gain privileges and access a socket used for authentication.
CVE-2009-3489	Photo editor installs a service with an insecure security descriptor, allowing users to stop or start the service, or execute commands as SYSTEM.
CVE-2009-3289	Library function copies a file to a new target and uses the source file's permissions for the target, which is incorrect when the source file is a symbolic link, which typically has 0777 permissions.
CVE-2009-0115	Device driver uses world-writable permissions for a socket file, allowing attackers to inject arbitrary commands.
CVE-2009-1073	LDAP server stores a cleartext password in a world-readable file.
CVE-2009-0141	Terminal emulator creates TTY devices with world-writable permissions, allowing an attacker to write to the terminals of other users.

CVE-2008-0662	VPN product stores user credentials in a registry key with "Everyone: Full Control" permissions, allowing attackers to steal the credentials.
CVE-2008-0322	Driver installs its device interface with "Everyone: Write" permissions.
CVE-2009-3939	Driver installs a file with world-writable permissions.
CVE-2009-3611	Product changes permissions to 0777 before deleting a backup; the permissions stay insecure for subsequent backups.
CVE-2007-6033	Product creates a share with "Everyone: Full Control" permissions, allowing arbitrary program execution.
CVE-2007-5544	Product uses "Everyone: Full Control" permissions for memory-mapped files (shared memory) in inter-process communication, allowing attackers to tamper with a session.
CVE-2005-4868	Database product uses read/write permissions for everyone for its shared memory, allowing theft of credentials.
CVE-2004-1714	Security product uses "Everyone: Full Control" permissions for its configuration files.
CVE-2001-0006	"Everyone: Full Control" permissions assigned to a mutex allows users to disable network connectivity.
CVE-2002-0969	Chain: database product contains buffer overflow that is only reachable through a .ini configuration file - which has "Everyone: Full Control" permissions.

Potential Mitigations

Phase: Implementation

When using a critical resource such as a configuration file, check to see if the resource has insecure permissions (such as being modifiable by any regular user), and generate an error or even exit the software if there is a possibility that the resource could have been modified by an unauthorized party.

Phase: Architecture and Design

Divide your application into anonymous, normal, privileged, and administrative areas. Reduce the attack surface by carefully defining distinct user groups, privileges, and/or roles. Map these against data, functionality, and the related resources. Then set the permissions accordingly. This will allow you to maintain more fine-grained control over your resources.

Phases: Implementation; Installation

During program startup, explicitly set the default permissions or umask to the most restrictive setting possible. Also set the appropriate permissions during program installation. This will prevent you from inheriting insecure permissions from any user who installs or runs the program.

Phase: System Configuration

For all configuration files, executables, and libraries, make sure that they are only readable and writable by the software's administrator.

Phase: Documentation

Do not suggest insecure configuration changes in your documentation, especially if those configurations can extend to resources and other software that are outside the scope of your own software.

Phase: Installation

Do not assume that the system administrator will manually change the configuration to the settings that you recommend in the manual.

Phase: Testing

Use tools and techniques that require manual (human) analysis, such as penetration testing, threat modeling, and interactive tools that allow the tester to record and modify an active session. These may be more effective than strictly automated techniques. This is especially the case with weaknesses that are related to design and business rules.

Phase: Testing

Use monitoring tools that examine the software's process as it interacts with the operating system and the network. This technique is useful in cases when source code is unavailable, if the software was not developed by you, or if you want to verify that the build phase did not introduce any new weaknesses. Examples include debuggers that directly attach to the running process; system-call tracing utilities such as truss (Solaris) and strace (Linux); system activity monitors such as FileMon, RegMon, Process Monitor, and other Sysinternals utilities (Windows); and sniffers and protocol analyzers that monitor network traffic.

Attach the monitor to the process and watch for library functions or system calls on OS resources such as files, directories, and shared memory. Examine the arguments to these calls to infer which permissions are being used.

Note that this technique is only useful for permissions issues related to system resources. It is not likely to detect application-level business rules that are related to permissions, such as if a user of a blog system marks a post as "private," but the blog system inadvertently marks it as "public."

Phases: Testing; System Configuration

Ensure that your software runs properly under the Federal Desktop Core Configuration (FDCC) or an equivalent hardening configuration guide, which many organizations use to limit the attack surface and potential risk of deployed software.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	275	Permission Issues	**Development Concepts (primary)699**
ChildOf	Weakness Class	668	Exposure of Resource to Wrong Sphere	**Research Concepts (primary)1000**
ChildOf	Category	753	2009 Top 25 - Porous Defenses	**Weaknesses in the 2009 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)750**
ChildOf	Category	803	2010 Top 25 - Porous Defenses	**Weaknesses in the 2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800**
RequiredBy	Compound Element: Composite	689	Permission Race Condition During Resource Copy	Research Concepts1000
ParentOf	Weakness Variant	276	Incorrect Default Permissions	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	277	Insecure Inherited Permissions	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	278	Insecure Preserved Inherited Permissions	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	279	Incorrect Execution-Assigned Permissions	**Research Concepts (primary)1000**
ParentOf	Weakness Base	281	Improper Preservation of Permissions	**Research Concepts (primary)1000**

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
232	Exploitation of Privilege/Trust	
1	Accessing Functionality Not Properly Constrained by ACLs	
17	Accessing, Modifying or Executing Executable Files	
60	Reusing Session IDs (aka Session Replay)	
61	Session Fixation	
62	Cross Site Request Forgery (aka Session Riding)	
122	Exploitation of Authorization	
180	Exploiting Incorrectly Configured Access Control Security Levels	
234	Hijacking a privileged process	

References

Mark Dowd, John McDonald and Justin Schuh. "The Art of Software Security Assessment". Chapter 9, "File Permissions." Page 495.. 1st Edition. Addison Wesley. 2006.

John Viega and Gary McGraw. "Building Secure Software". Chapter 8, "Access Control." Page 194.. 1st Edition. Addison-Wesley. 2002.

Maintenance Notes

The relationships between privileges, permissions, and actors (e.g. users and groups) need further refinement within the Research view. One complication is that these concepts apply to two different pillars, related to control of resources (CWE-664) and protection mechanism failures (CWE-396).

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
2008-09-08			Internal CWE Team
	new weakness-focused entry for Research view.		
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2009-01-12	CWE Content Team	MITRE	Internal
	updated Description, Likelihood of Exploit, Name, Potential Mitigations, Relationships		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Potential Mitigations, Related Attack Patterns		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Name		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Demonstrative Examples, Detection Factors, Modes of Introduction, Observed Examples, Potential Mitigations, References		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Relationships		
2010-04-05	CWE Content Team	MITRE	Internal
	updated Potential Mitigations, Related Attack Patterns		
Previous Entry Names			
Change Date	**Previous Entry Name**		
2009-01-12	Insecure Permission Assignment for Resource		
2009-05-27	Insecure Permission Assignment for Critical Resource		

BACK TO TOP

Exposure of System Data to Unauthorized Control Sphere

Risk

What might happen

System data can provide attackers with valuable insights on systems and services they are targeting - any type of system data, from service version to operating system fingerprints, can assist attackers to hone their attack, correlate data with known vulnerabilities or focus efforts on developing new attacks against specific technologies.

Cause

How does it happen

System data is read and subsequently exposed where it might be read by untrusted entities.

General Recommendations

How to avoid it

Consider the implications of exposure of the specified input, and expected level of access to the specified output. If not required, consider removing this code, or modifying exposed information to exclude potentially sensitive system data.

Source Code Examples

Java

Leaking Environment Variables in JSP Web-Page

```
String envVarValue = System.getenv(envVar);
if (envVarValue == null) {
    out.println("Environment variable is not defined:");
    out.println(System.getenv());
} else {
    //[..]
};
```

Potential Path Traversal

Risk

What might happen

An attacker could define any arbitrary file path for the application to use, potentially leading to:

- Stealing sensitive files, such as configuration or system files
- Overwriting files such as program binaries, configuration files, or system files
- Deleting critical files, causing a denial of service (DoS).

Cause

How does it happen

The application uses user input in the file path for accessing files on the application server's local disk. This enables an attacker to arbitrarily determine the file path.

General Recommendations

How to avoid it

1. Ideally, avoid depending on user input for file selection.
2. Validate all input, regardless of source. Validation should be based on a whitelist: accept only data fitting a specified structure, rather than reject bad patterns. Check for:
   - Data type
   - Size
   - Range
   - Format
   - Expected values
3. Accept user input only for the filename, not for the path and folders.
4. Ensure that file path is fully canonicalized.
5. Explicitly limit the application to using a designated folder that separate from the applications binary folder.
6. Restrict the privileges of the application's OS user to necessary files and folders. The application should not be able to write to the application binary folder, and should not read anything outside of the application folder and data folder.

Source Code Examples

CSharp

Using unvalidated user input as the file name may enable the user to access arbitrary files on the server local disk

```
public class PathTraversal
{
        private void foo(TextBox textbox1)
    {
                string fileNum = textbox1.Text;
                string path = "c:\files\file" + fileNum;
                FileStream f = new FileStream(path, FileMode.Open);
                byte[] output = new byte[10];
                f.Read(output,0, 10);
```

```
        }
}
```

Potentially hazardous characters are removed from the user input before use

```
public class PathTraversalFixed
{
        private void foo(TextBox textbox1)
    {
                string fileNum = textbox1.Text.Replace("\", "").Replace("..", "");
         string path = "c:\files\file" + fileNum;
                FileStream f = new FileStream(path, FileMode.Open);
                byte[] output = new byte[10];
                f.Read(output,0, 10);
        }
}
```

Java

Using unvalidated user input as the file name may enable the user to access arbitrary files on the server local disk

```
public class Absolute_Path_Traversal {
    public static void main(String[] args) {
        Scanner userInputScanner = new Scanner(System.in);
        System.out.print("\nEnter file name: ");
        String name = userInputScanner.nextLine();
        String path = "c:\files\file" + name;
        try {
            BufferedReader reader = new BufferedReader(new FileReader(path));
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

Potentially hazardous characters are removed from the user input before use

```
public class Absolute_Path_Traversal_Fixed {
    public static void main(String[] args) {
        Scanner userInputScanner = new Scanner(System.in);
        System.out.print("\nEnter file name: ");
        String name = userInputScanner.nextLine();
        name = name.replace("/", "").replace("..", "");
        String path = "c:\files\file" + name;
        try {
            BufferedReader reader = new BufferedReader(new FileReader(path));
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

Unchecked Return Value

Risk

What might happen

A program that does not check function return values could cause the application to enter an undefined state. This could lead to unexpected behavior and unintended consequences, including inconsistent data, system crashes or other error-based exploits.

Cause

How does it happen

The application calls a system function, but does not receive or check the result of this funciton. These functions often return error codes in the result, or share other status codes with it's caller. The application simply ignores this result value, losing this vital information.

General Recommendations

How to avoid it

- Always check the result of any called function that returns a value, and verify the result is an expected value.

- Ensure the calling function responds to all possible return values.

- Expect runtime errors and handle them gracefully. Explicitly define a mechanism for handling unexpected errors.

Source Code Examples

CPP

Unchecked Memory Allocation

```
buff = (char*) malloc(size);
strncpy(buff, source, size);
```

Safer Memory Allocation

```
buff = (char*) malloc(size+1);
if (buff==NULL) exit(1);

strncpy(buff, source, size);
buff[size] = '\0';
```

Use of sizeof() on a Pointer Type

Weakness ID: 467 *(Weakness Variant)* **Status:** Draft

Description

Description Summary

The code calls sizeof() on a malloced pointer type, which always returns the wordsize/8. This can produce an unexpected result if the programmer intended to determine how much memory has been allocated.

Time of Introduction

- Implementation

Applicable Platforms

Languages

C

C++

Common Consequences

Scope	Effect
Integrity	This error can often cause one to allocate a buffer that is much smaller than what is needed, leading to resultant weaknesses such as buffer overflows.

Likelihood of Exploit

High

Demonstrative Examples

Example 1

Care should be taken to ensure sizeof returns the size of the data structure itself, and not the size of the pointer to the data structure.

In this example, sizeof(foo) returns the size of the pointer.

(Bad Code)

Example Languages: **C and C++**

```
double *foo;
...
foo = (double *)malloc(sizeof(foo));
```

In this example, sizeof(*foo) returns the size of the data structure and not the size of the pointer.

(Good Code)

Example Languages: **C and C++**

```
double *foo;
...
foo = (double *)malloc(sizeof(*foo));
```

Example 2

This example defines a fixed username and password. The AuthenticateUser() function is intended to accept a username and a password from an untrusted user, and check to ensure that it matches the username and password. If the username and password match, AuthenticateUser() is intended to indicate that authentication succeeded.

(Bad Code)

```
/* Ignore CWE-259 (hard-coded password) and CWE-309 (use of password system for authentication) for this example. */

char *username = "admin";
char *pass = "password";

int AuthenticateUser(char *inUser, char *inPass) {
```

```
printf("Sizeof username = %d\n", sizeof(username));
printf("Sizeof pass = %d\n", sizeof(pass));

if (strncmp(username, inUser, sizeof(username))) {
printf("Auth failure of username using sizeof\n");
return(AUTH_FAIL);
}
/* Because of CWE-467, the sizeof returns 4 on many platforms and architectures. */
if (! strncmp(pass, inPass, sizeof(pass))) {
printf("Auth success of password using sizeof\n");
return(AUTH_SUCCESS);
}
else {
printf("Auth fail of password using sizeof\n");
return(AUTH_FAIL);
}
}

int main (int argc, char **argv)
{
int authResult;

if (argc < 3) {
ExitError("Usage: Provide a username and password");
}
authResult = AuthenticateUser(argv[1], argv[2]);
if (authResult != AUTH_SUCCESS) {
ExitError("Authentication failed");
}
else {
DoAuthenticatedTask(argv[1]);
}
}
```

In AuthenticateUser(), because sizeof() is applied to a parameter with an array type, the sizeof() call might return 4 on many modern architectures. As a result, the strncmp() call only checks the first four characters of the input password, resulting in a partial comparison (CWE-187), leading to improper authentication (CWE-287).

Because of the partial comparison, any of these passwords would still cause authentication to succeed for the "admin" user:

(Attack)

```
pass5
passABCDEFGH
passWORD
```

Because only 4 characters are checked, this significantly reduces the search space for an attacker, making brute force attacks more feasible.

The same problem also applies to the username, so values such as "adminXYZ" and "administrator" will succeed for the username.

Potential Mitigations

Phase: Implementation

Use expressions such as "sizeof(*pointer)" instead of "sizeof(pointer)", unless you intend to run sizeof() on a pointer type to gain some platform independence or if you are allocating a variable on the stack.

Other Notes

The use of sizeof() on a pointer can sometimes generate useful information. An obvious case is to find out the wordsize on a platform. More often than not, the appearance of sizeof(pointer) indicates a bug.

Weakness Ordinalities

Ordinality	Description
Primary	*(where the weakness exists independent of other weaknesses)*

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	465	Pointer Issues	**Development Concepts (primary)699**
ChildOf	Weakness Class	682	Incorrect Calculation	**Research Concepts (primary)1000**
ChildOf	Category	737	CERT C Secure Coding Section 03 - Expressions (EXP)	**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
CanPrecede	Weakness Base	131	Incorrect Calculation of Buffer Size	Research Concepts1000

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
CLASP			Use of sizeof() on a pointer type
CERT C Secure Coding	ARR01-C		Do not apply the sizeof operator to a pointer when taking the size of an array
CERT C Secure Coding	EXP01-C		Do not take the size of a pointer to determine the size of the pointed-to type

White Box Definitions

A weakness where code path has:

1. end statement that passes an identity of a dynamically allocated memory resource to a sizeof operator
2. start statement that allocates the dynamically allocated memory resource

References

Robert Seacord. "EXP01-A. Do not take the sizeof a pointer to determine the size of a type". <https://www.securecoding.cert.org/confluence/display/seccode/EXP01-A.+Do+not+take+the+sizeof+a+pointer+to+determine+the+size+of+a+type>.

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	CLASP		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-01		KDM Analytics	External
	added/updated white box definitions		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Relationships, Other Notes, Taxonomy Mappings, Weakness Ordinalities		
2008-11-24	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Relationships		

BACK TO TOP

Reliance on DNS Lookups in a Decision

Risk

What might happen

Relying on reverse DNS records, without verifying domain ownership via cryptographic certificates or protocols, is not a sufficient authentication mechanism. Basing any security decisions on the registered hostname could allow an external attacker to control the application flow. The attacker could possibly perform restricted operations, bypass access controls, and even spoof the user's identity, inject a bogus hostname into the security log, and possibly other logic attacks.

Cause

How does it happen

The application performs a reverse DNS resolution, based on the remote IP address, and performs a security check based on the returned hostname. However, it is relatively easy to spoof DNS names, or cause them to be misreported, depending on the context of the specific environment. If the remote server is controlled by the attacker, it can be configured to report a bogus hostname. Additionally, the attacker could also spoof the hostname if she controls the associated DNS server, or by attacking the legitimate DNS server, or by poisoning the server's DNS cache, or by modifying unprotected DNS traffic to the server. Regardless of the vector, a remote attacker can alter the detected network address, faking the authentication details.

General Recommendations

How to avoid it

- Do not rely on DNS records, network addresses, or system hostnames as a form of authentication, or any other security-related decision.
- Do not perform reverse DNS resolution over an unprotected protocol without record validation.
- Implement a proper authentication mechanism, such as passwords, cryptographic certificates, or public key digital signatures.
- Consider using proposed protocol extensions to cryptographically protect DNS, e.g. DNSSEC (though note the limited support and other drawbacks).

Source Code Examples

Java

Using Reverse DNS as Authentication

```
private boolean isInternalEmployee(ServletRequest req) {
      boolean isCompany = false;

      String ip = req.getRemoteAddr();
      InetAddress address = InetAddress.getByName(ip);

      if (address.getHostName().endsWith(COMPANYNAME)) {
            isCompany = true;
    }
      return isCompany;
```

```
}
```

Verify Authenticated User's Identity

```
private boolean isInternalEmployee(ServletRequest req) {
      boolean isCompany = false;

      Principal user = req.getUserPrincipal();
      if (user != null) {
      if (user.getName().startsWith(COMPANYDOMAIN + "\\")) {
            isCompany = true;
         }
     }
      return isCompany;
}
```

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

Use of sizeof() on a Pointer Type

Weakness ID: 467 *(Weakness Variant)* **Status:** Draft

Description

Description Summary

The code calls sizeof() on a malloced pointer type, which always returns the wordsize/8. This can produce an unexpected result if the programmer intended to determine how much memory has been allocated.

Time of Introduction

- Implementation

Applicable Platforms

Languages

C

C++

Common Consequences

Scope	Effect
Integrity	This error can often cause one to allocate a buffer that is much smaller than what is needed, leading to resultant weaknesses such as buffer overflows.

Likelihood of Exploit

High

Demonstrative Examples

Example 1

Care should be taken to ensure sizeof returns the size of the data structure itself, and not the size of the pointer to the data structure.

In this example, sizeof(foo) returns the size of the pointer.

(Bad Code)

Example Languages: **C and C++**

```
double *foo;
...
foo = (double *)malloc(sizeof(foo));
```

In this example, sizeof(*foo) returns the size of the data structure and not the size of the pointer.

(Good Code)

Example Languages: **C and C++**

```
double *foo;
...
foo = (double *)malloc(sizeof(*foo));
```

Example 2

This example defines a fixed username and password. The AuthenticateUser() function is intended to accept a username and a password from an untrusted user, and check to ensure that it matches the username and password. If the username and password match, AuthenticateUser() is intended to indicate that authentication succeeded.

(Bad Code)

```
/* Ignore CWE-259 (hard-coded password) and CWE-309 (use of password system for authentication) for this example. */

char *username = "admin";
char *pass = "password";

int AuthenticateUser(char *inUser, char *inPass) {
```

```
printf("Sizeof username = %d\n", sizeof(username));
printf("Sizeof pass = %d\n", sizeof(pass));

if (strncmp(username, inUser, sizeof(username))) {
printf("Auth failure of username using sizeof\n");
return(AUTH_FAIL);
}
/* Because of CWE-467, the sizeof returns 4 on many platforms and architectures. */
if (! strncmp(pass, inPass, sizeof(pass))) {
printf("Auth success of password using sizeof\n");
return(AUTH_SUCCESS);
}
else {
printf("Auth fail of password using sizeof\n");
return(AUTH_FAIL);
}
}

int main (int argc, char **argv)
{
int authResult;

if (argc < 3) {
ExitError("Usage: Provide a username and password");
}
authResult = AuthenticateUser(argv[1], argv[2]);
if (authResult != AUTH_SUCCESS) {
ExitError("Authentication failed");
}
else {
DoAuthenticatedTask(argv[1]);
}
}
```

In AuthenticateUser(), because sizeof() is applied to a parameter with an array type, the sizeof() call might return 4 on many modern architectures. As a result, the strncmp() call only checks the first four characters of the input password, resulting in a partial comparison (CWE-187), leading to improper authentication (CWE-287).

Because of the partial comparison, any of these passwords would still cause authentication to succeed for the "admin" user:

(Attack)

```
pass5
passABCDEFGH
passWORD
```

Because only 4 characters are checked, this significantly reduces the search space for an attacker, making brute force attacks more feasible.

The same problem also applies to the username, so values such as "adminXYZ" and "administrator" will succeed for the username.

Potential Mitigations

Phase: Implementation

Use expressions such as "sizeof(*pointer)" instead of "sizeof(pointer)", unless you intend to run sizeof() on a pointer type to gain some platform independence or if you are allocating a variable on the stack.

Other Notes

The use of sizeof() on a pointer can sometimes generate useful information. An obvious case is to find out the wordsize on a platform. More often than not, the appearance of sizeof(pointer) indicates a bug.

Weakness Ordinalities

Ordinality	Description
Primary	*(where the weakness exists independent of other weaknesses)*

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	465	Pointer Issues	**Development Concepts (primary)699**
ChildOf	Weakness Class	682	Incorrect Calculation	**Research Concepts (primary)1000**
ChildOf	Category	737	CERT C Secure Coding Section 03 - Expressions (EXP)	**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
CanPrecede	Weakness Base	131	Incorrect Calculation of Buffer Size	Research Concepts1000

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
CLASP			Use of sizeof() on a pointer type
CERT C Secure Coding	ARR01-C		Do not apply the sizeof operator to a pointer when taking the size of an array
CERT C Secure Coding	EXP01-C		Do not take the size of a pointer to determine the size of the pointed-to type

White Box Definitions

A weakness where code path has:

1. end statement that passes an identity of a dynamically allocated memory resource to a sizeof operator
2. start statement that allocates the dynamically allocated memory resource

References

Robert Seacord. "EXP01-A. Do not take the sizeof a pointer to determine the size of a type". <https://www.securecoding.cert.org/confluence/display/seccode/EXP01-A.+Do+not+take+the+sizeof+a+pointer+to+determine+the+size+of+a+type>.

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	CLASP		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-01		KDM Analytics	External
	added/updated white box definitions		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Relationships, Other Notes, Taxonomy Mappings, Weakness Ordinalities		
2008-11-24	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Relationships		

BACK TO TOP

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

TOCTOU

Risk

What might happen

At best, a Race Condition may cause errors in accuracy, overidden values or unexpected behavior that may result in denial-of-service. At worst, it may allow attackers to retrieve data or bypass security processes by replaying a controllable Race Condition until it plays out in their favor.

Cause

How does it happen

Race Conditions occur when a public, single instance of a resource is used by multiple concurrent logical processes. If the these logical processes attempt to retrieve and update the resource without a timely management system, such as a lock, a Race Condition will occur.

An example for when a Race Condition occurs is a resource that may return a certain value to a process for further editing, and then updated by a second process, resulting in the original process' data no longer being valid. Once the original process edits and updates the incorrect value back into the resource, the second process' update has been overwritten and lost.

General Recommendations

How to avoid it

When sharing resources between concurrent processes across the application ensure that these resources are either thread-safe, or implement a locking mechanism to ensure expected concurrent activity.

Source Code Examples

Java

Different Threads Increment and Decrement The Same Counter Repeatedly, Resulting in a Race Condition

```
	public static int counter = 0;
	public static void start() throws InterruptedException {
		incrementCounter ic;
		decrementCounter dc;
		while(counter == 0) {
			counter = 0;
			ic = new incrementCounter();
			dc = new decrementCounter();
			ic.start();
			dc.start();
			ic.join();
			dc.join();
		}
		System.out.println(counter); //Will stop and return either -1 or 1 due to race
condition over counter
	}

	public static class incrementCounter extends Thread {
	    public void run() {
		counter++;
	    }
```

```
    }

    public static class decrementCounter extends Thread {
        public void run() {
           counter--;
        }
    }
```

Different Threads Increment and Decrement The Same Thread-Safe Counter Repeatedly, Never Resulting in a Race Condition

```
    public static int counter = 0;
    public static Object lock = new Object();

    public static void start() throws InterruptedException {
           incrementCounter ic;
           decrementCounter dc;
           while(counter == 0) { // because of proper locking, this condition is never false
                  counter = 0;
                  ic = new incrementCounter();
                  dc = new decrementCounter();
                  ic.start();
                  dc.start();
                  ic.join();
                  dc.join();
           }
           System.out.println(counter); // Never reached
    }

    public static class incrementCounter extends Thread {
        public void run() {
           synchronized (lock) {
                  counter++;
           }
        }
    }

    public static class decrementCounter extends Thread {
        public void run() {
           synchronized (lock) {
                  counter--;
           }
        }
    }
```

Scanned Languages

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