vul_files_43 Scan Report

Project Name	vul_files_43
Scan Start	Tuesday, January 7, 2025 11:33:16 PM
Preset	Checkmarx Default
Scan Time	03h:14m:18s
Lines Of Code Scanned	297599
Files Scanned	88
Report Creation Time	Wednesday, January 8, 2025 9:51:47 AM
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044
Team	CxServer
Checkmarx Version	8.7.0
Scan Type	Full
Source Origin	LocalPath
Density	3/1000 (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

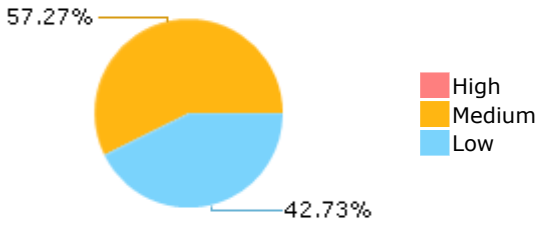
Results Limit

Results limit per query was set to 50

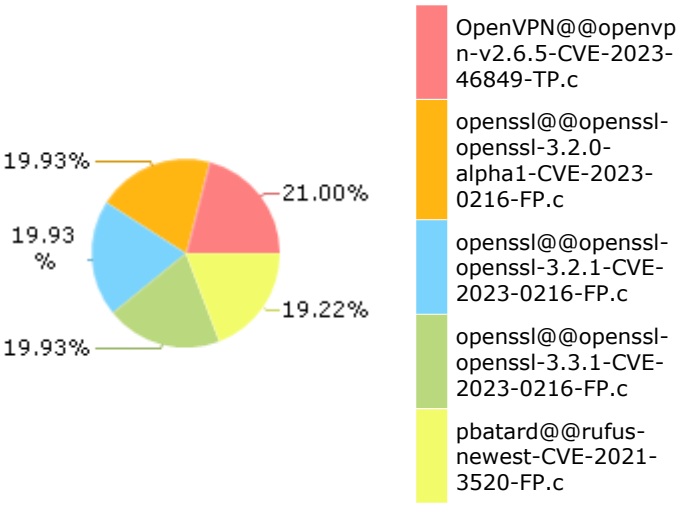
Selected Queries

Selected queries are listed in Result Summary

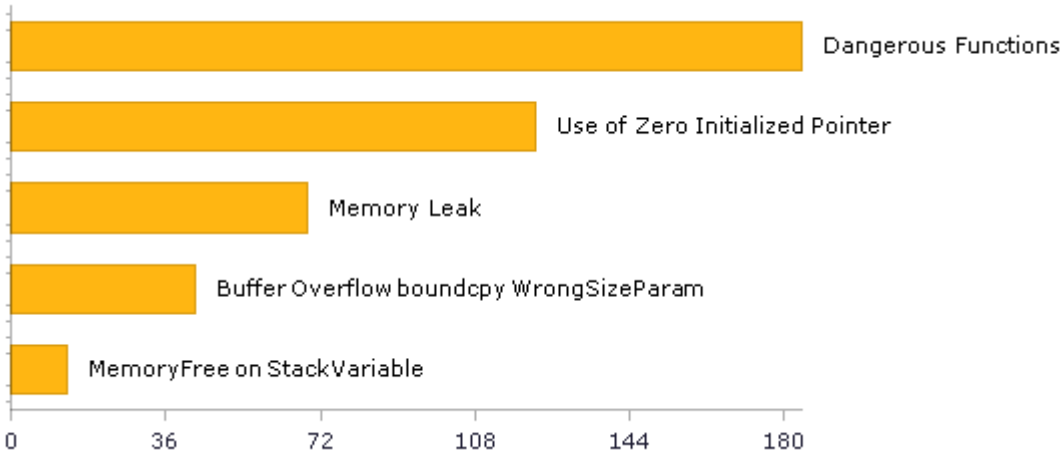
Result Summary



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	254	56
A2-Broken Authentication	App. Specific	EASY	COMMON	AVERAGE	SEVERE	App. Specific	47	47
A3-Sensitive Data Exposure	App. Specific	AVERAGE	WIDESPREAD	AVERAGE	SEVERE	App. Specific	6	6
A4-XML External Entities (XXE)	App. Specific	AVERAGE	COMMON	EASY	SEVERE	App. Specific	0	0
A5-Broken Access Control*	App. Specific	AVERAGE	COMMON	AVERAGE	SEVERE	App. Specific	0	0
A6-Security Misconfiguration	App. Specific	EASY	WIDESPREAD	EASY	MODERATE	App. Specific	0	0
A7-Cross-Site Scripting (XSS)	App. Specific	EASY	WIDESPREAD	EASY	MODERATE	App. Specific	0	0
A8-Insecure Deserialization	App. Specific	DIFFICULT	COMMON	AVERAGE	SEVERE	App. Specific	0	0
A9-Using Components with Known Vulnerabilities*	App. Specific	AVERAGE	WIDESPREAD	AVERAGE	MODERATE	App. Specific	184	184
A10-Insufficient Logging & Monitoring	App. Specific	AVERAGE	WIDESPREAD	DIFFICULT	MODERATE	App. Specific	0	0

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

Scan Summary - OWASP Top 10 2013

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

Category	Threat Agent	Attack Vectors	Weakness Prevalence	Weakness Detectability	Technical Impact	Business Impact	Issues Found	Best Fix Locations
A1-Injection	EXTERNAL, INTERNAL, ADMIN USERS	EASY	COMMON	AVERAGE	SEVERE	ALL DATA	0	0
A2-Broken Authentication and Session Management	EXTERNAL, INTERNAL USERS	AVERAGE	WIDESPREAD	AVERAGE	SEVERE	AFFECTED DATA AND FUNCTIONS	0	0
A3-Cross-Site Scripting (XSS)	EXTERNAL, INTERNAL, ADMIN USERS	AVERAGE	VERY WIDESPREAD	EASY	MODERATE	AFFECTED DATA AND SYSTEM	0	0
A4-Insecure Direct Object References	SYSTEM USERS	EASY	COMMON	EASY	MODERATE	EXPOSED DATA	0	0
A5-Security Misconfiguration	EXTERNAL, INTERNAL, ADMIN USERS	EASY	COMMON	EASY	MODERATE	ALL DATA AND SYSTEM	0	0
A6-Sensitive Data Exposure	EXTERNAL, INTERNAL, ADMIN USERS, USERS BROWSERS	DIFFICULT	UNCOMMON	AVERAGE	SEVERE	EXPOSED DATA	2	2
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	184	184
A10-Unvalidated Redirects and Forwards	USERS BROWSERS	AVERAGE	WIDESPREAD	DIFFICULT	MODERATE	AFFECTED DATA AND FUNCTIONS	0	0

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

Scan Summary - PCI DSS v3.2

Category	Issues Found	Best Fix Locations
PCI DSS (3.2) - 6.5.1 - Injection flaws - particularly SQL injection	0	0
PCI DSS (3.2) - 6.5.2 - Buffer overflows	55	55
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.	8	8
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.	1	1
Configuration Management	Organizations must: (i) establish and maintain baseline configurations and inventories of organizational information systems (including hardware, software, firmware, and documentation) throughout the respective system development life cycles; and (ii) establish and enforce security configuration settings for information technology products employed in organizational information systems.	0	0
Identification And Authentication*	Organizations must identify information system users, processes acting on behalf of users, or devices and authenticate (or verify) the identities of those users, processes, or devices, as a prerequisite to allowing access to organizational information systems.	43	43
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.	6	6
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.	4	4

***** 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)	47	47
AC-4 Information Flow Enforcement (P1)	0	0
AC-6 Least Privilege (P1)	0	0
AU-9 Protection of Audit Information (P1)	0	0
CM-6 Configuration Settings (P2)	0	0
IA-5 Authenticator Management (P1)	0	0
IA-6 Authenticator Feedback (P2)	0	0
IA-8 Identification and Authentication (Non-Organizational Users) (P1)	0	0
SC-12 Cryptographic Key Establishment and Management (P1)	0	0
SC-13 Cryptographic Protection (P1)	0	0
SC-17 Public Key Infrastructure Certificates (P1)	0	0
SC-18 Mobile Code (P2)	0	0
SC-23 Session Authenticity (P1)*	0	0
SC-28 Protection of Information at Rest (P1)	8	8
SC-4 Information in Shared Resources (P1)	2	2
SC-5 Denial of Service Protection (P1)*	402	111
SC-8 Transmission Confidentiality and Integrity (P1)	0	0
SI-10 Information Input Validation (P1)*	22	22
SI-11 Error Handling (P2)*	60	60
SI-15 Information Output Filtering (P0)	0	0
SI-16 Memory Protection (P1)	1	1

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

Scan Summary - OWASP Mobile Top 10 2016

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

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

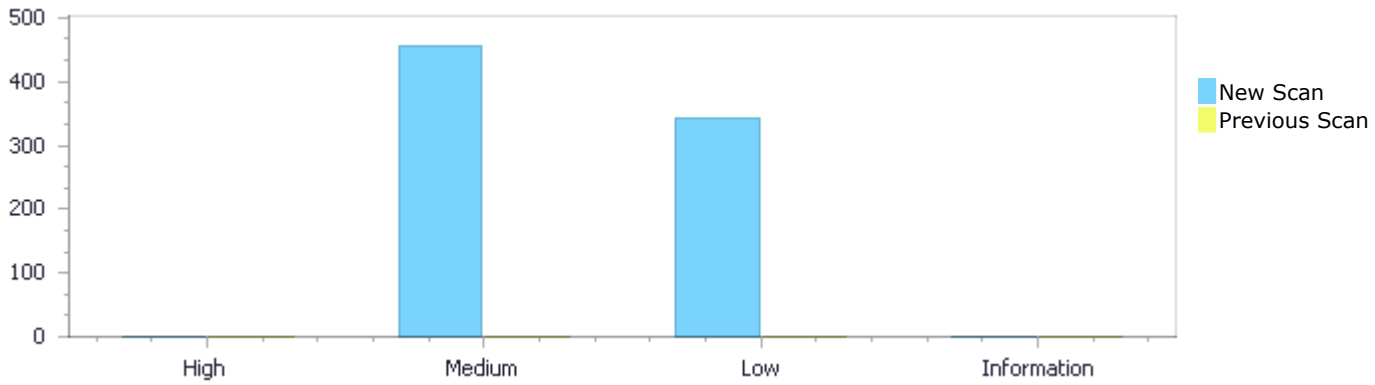
Scan Summary - Custom

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

Results Distribution By Status

First scan of the project

	High	Medium	Low	Information	Total
New Issues	0	457	341	0	798
Recurrent Issues	0	0	0	0	0
Total	0	457	341	0	798
Fixed Issues	0	0	0	0	0



Results Distribution By State

	High	Medium	Low	Information	Total
Confirmed	0	0	0	0	0
Not Exploitable	0	0	0	0	0
To Verify	0	457	341	0	798
Urgent	0	0	0	0	0
Proposed Not Exploitable	0	0	0	0	0
Total	0	457	341	0	798

Result Summary

Vulnerability Type	Occurrences	Severity
Dangerous Functions	184	Medium
Use of Zero Initialized Pointer	122	Medium
Memory Leak	69	Medium
Buffer Overflow boundcpy WrongSizeParam	43	Medium
MemoryFree on StackVariable	13	Medium

Wrong Size t Allocation	12	Medium
Char Overflow	6	Medium
Integer Overflow	4	Medium
Heap Inspection	2	Medium
Buffer Overflow AddressOfLocalVarReturned	1	Medium
Buffer Overflow Loops	1	Medium
NULL Pointer Dereference	209	Low
Unchecked Return Value	60	Low
Improper Resource Access Authorization	39	Low
Unchecked Array Index	12	Low
Incorrect Permission Assignment For Critical Resources	8	Low
Information Exposure Through Comments	4	Low
Use of Insufficiently Random Values	4	Low
Sizeof Pointer Argument	2	Low
TOCTOU	2	Low
Arithmenic Operation On Boolean	1	Low

10 Most Vulnerable Files

High and Medium Vulnerabilities

File Name	Issues Found
OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c	39
OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c	39
OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c	39
pbatard@@rufus-newest-CVE-2021-3520-FP.c	37
openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	36
openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	34
openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	21
openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	21
OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c	17
openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c	17

Scan Results Details

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=1020051&projectid=20044&pathid=369
Status	New

The dangerous function, memcpy, was found in use at line 243 in openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Line	252	252
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Method static int ssl_session_memcpy(unsigned char *dst, size_t *pdstlen,

```
....
252.      memcpy(dst, src->data, src->length);
```

Dangerous Functions\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=370
Status	New

The dangerous function, memcpy, was found in use at line 92 in openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination

File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	253	253
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
253.                memcpy(q++, p, sizeof(*q));
```

Dangerous Functions\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=371
Status	New

The dangerous function, memcpy, was found in use at line 92 in openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	261	261
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
261.                memcpy(q++, p, sizeof(*q));
```

Dangerous Functions\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=372
Status	New

The dangerous function, memcpy, was found in use at line 92 in openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	411	411
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
411.                    memcpy(ctx->oiv, iv, n);
```

Dangerous Functions\Path 5:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=373
Status New

The dangerous function, memcpy, was found in use at line 92 in openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	412	412
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
412.                memcpy(ctx->iv, ctx->oiv, n);
```

Dangerous Functions\Path 6:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=374
Status New

The dangerous function, memcpy, was found in use at line 92 in openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	424	424
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
424.                    memcpy(ctx->iv, iv, n);
```

Dangerous Functions\Path 7:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=375
Status New

The dangerous function, memcpy, was found in use at line 567 in openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	615	615
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_EncryptDecryptUpdate(EVP_CIPHER_CTX *ctx,

```
....
615.                memcpy(&(ctx->buf[i]), in, inl);
```

Dangerous Functions\Path 8:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=376
Status New

The dangerous function, memcpy, was found in use at line 567 in openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	633	633
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_EncryptDecryptUpdate(EVP_CIPHER_CTX *ctx,

```
....
633.                memcpy(&(ctx->buf[i]), in, j);
```

Dangerous Functions\Path 9:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=377
Status New

The dangerous function, memcpy, was found in use at line 567 in openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	652	652
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_EncryptDecryptUpdate(EVP_CIPHER_CTX *ctx,

```
....
652.            memcpy(ctx->buf, &(in[inl]), i);
```

Dangerous Functions\Path 10:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=378
Status New

The dangerous function, memcpy, was found in use at line 806 in openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	907	907
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method int EVP_DecryptUpdate(EVP_CIPHER_CTX *ctx, unsigned char *out, int *outl,

```
....
907.            memcpy(out, ctx->final, b);
```

Dangerous Functions\Path 11:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=379
Status New

The dangerous function, memcpy, was found in use at line 806 in openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	923	923
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method int EVP_DecryptUpdate(EVP_CIPHER_CTX *ctx, unsigned char *out, int *outl,

```
....
923.            memcpy(ctx->final, &out[*outl], b);
```

Dangerous Functions\Path 12:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=380
Status New

The dangerous function, memcpy, was found in use at line 1458 in openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	1503	1503
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method int EVP_CIPHER_CTX_copy(EVP_CIPHER_CTX *out, const EVP_CIPHER_CTX *in)

```
....
1503.       memcpy(out, in, sizeof(*out));
```

Dangerous Functions\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=381
Status	New

The dangerous function, memcpy, was found in use at line 1458 in openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	1511	1511
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method int EVP_CIPHER_CTX_copy(EVP_CIPHER_CTX *out, const EVP_CIPHER_CTX *in)

```
....
1511.           memcpy(out->cipher_data, in->cipher_data, in->cipher-
>ctx_size);
```

Dangerous Functions\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=382
Status	New

The dangerous function, memcpy, was found in use at line 746 in openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c
Line	761	761
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c
Method static void extension_append(unsigned int version,

```
....
761.      memcpy(serverinfo + contextoff, extension, extension_length);
```

Dangerous Functions\Path 15:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=383
Status New

The dangerous function, memcpy, was found in use at line 764 in openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c
Line	810	810
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c
Method int SSL_CTX_use_serverinfo_ex(SSL_CTX *ctx, unsigned int version,

```
....
810.      memcpy(ctx->cert->key->serverinfo, serverinfo,
serverinfo_length);
```

Dangerous Functions\Path 16:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=384

Status	New

The dangerous function, memcpy, was found in use at line 243 in openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c
Line	252	252
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c
Method static int ssl_session_memcpy(unsigned char *dst, size_t *pdstlen,

```
....
252.      memcpy(dst, src->data, src->length);
```

Dangerous Functions\Path 17:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=385
Status	New

The dangerous function, memcpy, was found in use at line 92 in openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	253	253
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
253.              memcpy(q++, p, sizeof(*q));
```

Dangerous Functions\Path 18:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20

	044&pathid=386
Status	New

The dangerous function, memcpy, was found in use at line 92 in openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	261	261
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
261.                memcpy(q++, p, sizeof(*q));
```

Dangerous Functions\Path 19:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=387
Status	New

The dangerous function, memcpy, was found in use at line 92 in openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	411	411
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
411.                    memcpy(ctx->oiv, iv, n);
```

Dangerous Functions\Path 20:

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

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=388
Status	New

The dangerous function, memcpy, was found in use at line 92 in openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	412	412
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
412.                memcpy(ctx->iv, ctx->oiv, n);
```

Dangerous Functions\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=389
Status	New

The dangerous function, memcpy, was found in use at line 92 in openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	424	424
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
424.                    memcpy(ctx->iv, iv, n);
```

Dangerous Functions\Path 22:

Severity	Medium
Result State	To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=390
Status	New

The dangerous function, memcpy, was found in use at line 567 in openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	615	615
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method static int evp_EncryptDecryptUpdate(EVP_CIPHER_CTX *ctx,

```
....
615.                memcpy(&(ctx->buf[i]), in, inl);
```

Dangerous Functions\Path 23:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=391
Status	New

The dangerous function, memcpy, was found in use at line 567 in openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	633	633
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method static int evp_EncryptDecryptUpdate(EVP_CIPHER_CTX *ctx,

```
....
633.                memcpy(&(ctx->buf[i]), in, j);
```

Dangerous Functions\Path 24:

Severity	Medium

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=392
Status	New

The dangerous function, memcpy, was found in use at line 567 in openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	652	652
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method static int evp_EncryptDecryptUpdate(EVP_CIPHER_CTX *ctx,

```
....
652.            memcpy(ctx->buf, &(in[inl]), i);
```

Dangerous Functions\Path 25:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=393
Status	New

The dangerous function, memcpy, was found in use at line 806 in openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	907	907
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method int EVP_DecryptUpdate(EVP_CIPHER_CTX *ctx, unsigned char *out, int *outl,

```
....
907.            memcpy(out, ctx->final, b);
```

Dangerous Functions\Path 26:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=394
Status	New

The dangerous function, memcpy, was found in use at line 806 in openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	923	923
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method int EVP_DecryptUpdate(EVP_CIPHER_CTX *ctx, unsigned char *out, int *outl,

```
....
923.          memcpy(ctx->final, &out[*outl], b);
```

Dangerous Functions\Path 27:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=395
Status	New

The dangerous function, memcpy, was found in use at line 1458 in openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	1503	1503
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method int EVP_CIPHER_CTX_copy(EVP_CIPHER_CTX *out, const EVP_CIPHER_CTX *in)

```
....
1503.      memcpy(out, in, sizeof(*out));
```

Dangerous Functions\Path 28:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=396
Status	New

The dangerous function, memcpy, was found in use at line 1458 in openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	1511	1511
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method int EVP_CIPHER_CTX_copy(EVP_CIPHER_CTX *out, const EVP_CIPHER_CTX *in)

```
....
1511.          memcpy(out->cipher_data, in->cipher_data, in->cipher-
>ctx_size);
```

Dangerous Functions\Path 29:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=397
Status	New

The dangerous function, memcpy, was found in use at line 746 in openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c
Line	761	761
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c
Method static void extension_append(unsigned int version,

```
....
761.      memcpy(serverinfo + contextoff, extension, extension_length);
```

Dangerous Functions\Path 30:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=398
Status	New

The dangerous function, memcpy, was found in use at line 764 in openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c
Line	810	810
Object	memcpy	memcpy

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c
Method int SSL_CTX_use_serverinfo_ex(SSL_CTX *ctx, unsigned int version,

```
....
810.      memcpy(ctx->cert->key->serverinfo, serverinfo,
serverinfo_length);
```

Dangerous Functions\Path 31:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=399
Status	New

The dangerous function, memcpy, was found in use at line 243 in openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c
Line	252	252
Object	memcpy	memcpy

Code Snippet

File Name	openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c
Method	static int ssl_session_memcpy(unsigned char *dst, size_t *pdstlen,

```
....
252.      memcpy(dst, src->data, src->length);
```

Dangerous Functions\Path 32:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=400
Status	New

The dangerous function, memcpy, was found in use at line 1330 in OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Line	1447	1447
Object	memcpy	memcpy

Code Snippet

File Name	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Method	ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1447.          memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 33:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=401
Status	New

The dangerous function, memcpy, was found in use at line 1330 in OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Line	1448	1448
Object	memcpy	memcpy

Code Snippet	
File Name	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Method	ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1448.          memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 34:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=402
Status	New

The dangerous function, memcpy, was found in use at line 1330 in OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c
Line	1447	1447
Object	memcpy	memcpy

Code Snippet	
File Name	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c
Method	ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1447.          memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 35:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=403
Status	New

The dangerous function, memcpy, was found in use at line 1330 in OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c

Line	1448	1448
Object	memcpy	memcpy

Code Snippet
File Name OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1448.           memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 36:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=404
Status	New

The dangerous function, memcpy, was found in use at line 1332 in OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Line	1449	1449
Object	memcpy	memcpy

Code Snippet
File Name OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1449.           memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 37:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=405
Status	New

The dangerous function, memcpy, was found in use at line 1332 in OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination

File	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Line	1450	1450
Object	memcpy	memcpy

Code Snippet
File Name OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1450.           memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 38:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=406
Status	New

The dangerous function, memcpy, was found in use at line 1332 in OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Line	1449	1449
Object	memcpy	memcpy

Code Snippet
File Name OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1449.           memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 39:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=407
Status	New

The dangerous function, memcpy, was found in use at line 1332 in OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Line	1450	1450
Object	memcpy	memcpy

Code Snippet
File Name OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1450.           memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 40:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=408
Status	New

The dangerous function, memcpy, was found in use at line 1332 in OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Line	1449	1449
Object	memcpy	memcpy

Code Snippet
File Name OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1449.           memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 41:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=409
Status	New

The dangerous function, memcpy, was found in use at line 1332 in OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Line	1450	1450
Object	memcpy	memcpy

Code Snippet
File Name OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1450.           memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 42:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=410
Status New

The dangerous function, memcpy, was found in use at line 1332 in OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Line	1449	1449
Object	memcpy	memcpy

Code Snippet
File Name OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1449.           memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 43:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20

	044&pathid=411
Status	New

The dangerous function, memcpy, was found in use at line 1332 in OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Line	1450	1450
Object	memcpy	memcpy

Code Snippet
File Name OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1450.           memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 44:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=412
Status	New

The dangerous function, memcpy, was found in use at line 1332 in OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Line	1449	1449
Object	memcpy	memcpy

Code Snippet
File Name OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1449.           memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 45:

Severity	Medium

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=413
Status	New

The dangerous function, memcpy, was found in use at line 1332 in OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Line	1450	1450
Object	memcpy	memcpy

Code Snippet
File Name OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1450.           memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 46:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=414
Status	New

The dangerous function, memcpy, was found in use at line 3221 in OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Line	3325	3325
Object	memcpy	memcpy

Code Snippet
File Name OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Method do_init_crypto_tls(struct context *c, const unsigned int flags)

```
....
3325.       memcpy(to.remote_cert_ku, options->remote_cert_ku,
sizeof(to.remote_cert_ku));
```

Dangerous Functions\Path 47:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=415
Status	New

The dangerous function, memcpy, was found in use at line 3221 in OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Line	3332	3332
Object	memcpy	memcpy

Code Snippet
File Name OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Method do_init_crypto_tls(struct context *c, const unsigned int flags)

```
....
3332.       memcpy(to.x509_username_field, options->x509_username_field,
sizeof(to.x509_username_field));
```

Dangerous Functions\Path 48:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=416
Status	New

The dangerous function, memcpy, was found in use at line 1493 in OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	1610	1610
Object	memcpy	memcpy

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1610.          memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 49:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=417
Status	New

The dangerous function, memcpy, was found in use at line 1493 in OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	1611	1611
Object	memcpy	memcpy

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1611.          memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Dangerous Functions\Path 50:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=418
Status	New

The dangerous function, memcpy, was found in use at line 1678 in OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Line	1688	1688
Object	memcpy	memcpy

Code Snippet	
File Name	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Method	key_ctx_update_implicit_iv(struct key_ctx *ctx, uint8_t *key, size_t key_len)

```
....
1688.          memcpy(ctx->implicit_iv, key, impl_iv_len);
```

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=1020051&projectid=20044&pathid=624
Status	New

The variable declared in serverinfo at openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c in line 832 is not initialized when it is used by serverinfo at openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c in line 832.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c
Line	834	920
Object	serverinfo	serverinfo

Code Snippet	
File Name	openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c
Method	int SSL_CTX_use_serverinfo_file(SSL_CTX *ctx, const char *file)

```
....
834.      unsigned char *serverinfo = NULL;
....
920.          serverinfo = tmp;
```

Use of Zero Initialized Pointer\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=625
Status	New

The variable declared in serverinfo at openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c in line 832 is not initialized when it is used by serverinfo at openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c in line 832.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c
Line	834	920
Object	serverinfo	serverinfo

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c
Method int SSL_CTX_use_serverinfo_file(SSL_CTX *ctx, const char *file)

```
....
834.      unsigned char *serverinfo = NULL;
....
920.          serverinfo = tmp;
```

Use of Zero Initialized Pointer\Path 3:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=626
Status New

The variable declared in current_comment_before_key at openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c in line 235 is not initialized when it is used by comment_before_key at openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c in line 121.

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	484	123
Object	current_comment_before_key	comment_before_key

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
484.      current_comment_before_key = NULL;
```

File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method store (econf_file *ef, const char *group, const char *key,

```
....
123.         const char *comment_before_key, const char
*comment_after_value,
```

Use of Zero Initialized Pointer\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=627
Status	New

The variable declared in current_comment_before_key at openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c in line 235 is not initialized when it is used by comment_before_key at openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c in line 121.

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	405	123
Object	current_comment_before_key	comment_before_key

Code Snippet

File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c

Method read_file(econf_file *ef, const char *file,

```
....
405.         current_comment_before_key = NULL;
```

File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c

Method store (econf_file *ef, const char *group, const char *key,

```
....
123.         const char *comment_before_key, const char
*comment_after_value,
```

Use of Zero Initialized Pointer\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=628
Status	New

The variable declared in current_comment_before_key at openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c in line 235 is not initialized when it is used by comment_before_key at openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c in line 121.

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	240	123
Object	current_comment_before_key	comment_before_key

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
240.    char *current_comment_before_key = NULL;
```

File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method store (econf_file *ef, const char *group, const char *key,

```
....
123.         const char *comment_before_key, const char
*comment_after_value,
```

Use of Zero Initialized Pointer\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=629
Status	New

The variable declared in current_comment_after_value at openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c in line 235 is not initialized when it is used by comment_after_value at openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c in line 121.

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	407	123
Object	current_comment_after_value	comment_after_value

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
407.        current_comment_after_value = NULL;
```

File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c

Method store (econf_file *ef, const char *group, const char *key,

```
....
123.         const char *comment_before_key, const char
*comment_after_value,
```

Use of Zero Initialized Pointer\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=630
Status	New

The variable declared in current_comment_after_value at openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c in line 235 is not initialized when it is used by comment_after_value at openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c in line 121.

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	486	123
Object	current_comment_after_value	comment_after_value

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
486.      current_comment_after_value = NULL;
```

File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c

Method store (econf_file *ef, const char *group, const char *key,

```
....
123.         const char *comment_before_key, const char
*comment_after_value,
```

Use of Zero Initialized Pointer\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=631
Status	New

The variable declared in current_comment_after_value at openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c in line 235 is not initialized when it is used by comment_after_value at openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c in line 121.

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	241	123
Object	current_comment_after_value	comment_after_value

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
241.    char *current_comment_after_value = NULL;
```

File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method store (econf_file *ef, const char *group, const char *key,

```
....
123.         const char *comment_before_key, const char
*comment_after_value,
```

Use of Zero Initialized Pointer\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=632
Status	New

The variable declared in current_comment_before_key at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 242 is not initialized when it is used by comment_before_key at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 121.

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	432	123
Object	current_comment_before_key	comment_before_key

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
432.        current_comment_before_key = NULL;
```

File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c

Method	store (econf_file *ef, const char *group, const char *key,

```
....
123.         const char *comment_before_key, const char
*comment_after_value,
```

Use of Zero Initialized Pointer\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=633
Status	New

The variable declared in current_comment_before_key at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 242 is not initialized when it is used by comment_before_key at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 121.

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	512	123
Object	current_comment_before_key	comment_before_key

Code Snippet

File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c

Method read_file(econf_file *ef, const char *file,

```
....
512.      current_comment_before_key = NULL;
```

File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c

Method store (econf_file *ef, const char *group, const char *key,

```
....
123.         const char *comment_before_key, const char
*comment_after_value,
```

Use of Zero Initialized Pointer\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=634
Status	New

The variable declared in current_comment_before_key at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 242 is not initialized when it is used by comment_before_key at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 121.

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	362	123
Object	current_comment_before_key	comment_before_key

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
362.        current_comment_before_key = NULL;
```

File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method store (econf_file *ef, const char *group, const char *key,

```
....
123.         const char *comment_before_key, const char
*comment_after_value,
```

Use of Zero Initialized Pointer\Path 12:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=635
Status New

The variable declared in current_comment_before_key at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 242 is not initialized when it is used by comment_before_key at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 121.

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	247	123
Object	current_comment_before_key	comment_before_key

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
247.    char *current_comment_before_key = NULL;
```

File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c

Method store (econf_file *ef, const char *group, const char *key,

```
....
123.         const char *comment_before_key, const char
*comment_after_value,
```

Use of Zero Initialized Pointer\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=636
Status	New

The variable declared in current_comment_after_value at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 242 is not initialized when it is used by comment_after_value at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 121.

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	514	123
Object	current_comment_after_value	comment_after_value

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
514.      current_comment_after_value = NULL;
```

File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method store (econf_file *ef, const char *group, const char *key,

```
....
123.         const char *comment_before_key, const char
*comment_after_value,
```

Use of Zero Initialized Pointer\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=637
Status	New

The variable declared in current_comment_after_value at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 242 is not initialized when it is used by comment_after_value at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 121.

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	434	123
Object	current_comment_after_value	comment_after_value

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
434.        current_comment_after_value = NULL;
```

File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method store (econf_file *ef, const char *group, const char *key,

```
....
123.         const char *comment_before_key, const char
*comment_after_value,
```

Use of Zero Initialized Pointer\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=638
Status	New

The variable declared in current_comment_after_value at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 242 is not initialized when it is used by comment_after_value at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 121.

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	364	123
Object	current_comment_after_value	comment_after_value

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
364.        current_comment_after_value = NULL;
```

File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c

Method	store (econf_file *ef, const char *group, const char *key,

```
....
123.         const char *comment_before_key, const char
*comment_after_value,
```

Use of Zero Initialized Pointer\Path 16:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=639
Status	New

The variable declared in current_comment_after_value at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 242 is not initialized when it is used by comment_after_value at openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c in line 121.

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	248	123
Object	current_comment_after_value	comment_after_value

Code Snippet

File Name	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method	read_file(econf_file *ef, const char *file,

```
....
248.    char *current_comment_after_value = NULL;
```

File Name	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method	store (econf_file *ef, const char *group, const char *key,

```
....
123.         const char *comment_before_key, const char
*comment_after_value,
```

Use of Zero Initialized Pointer\Path 17:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=640
Status	New

The variable declared in up at OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c in line 2335 is not initialized when it is used by up at OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c in line 2335.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Line	2343	2439
Object	up	up

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Method key_method_2_read(struct buffer *buf, struct tls_multi *multi, struct tls_session *session)

```
....
2343.       struct user_pass *up = NULL;
....
2439.       secure_memzero(up, sizeof(*up));
```

Use of Zero Initialized Pointer\Path 18:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=641
Status New

The variable declared in up at OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c in line 2335 is not initialized when it is used by up at OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c in line 2335.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Line	2343	2416
Object	up	up

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Method key_method_2_read(struct buffer *buf, struct tls_multi *multi, struct tls_session *session)

```
....
2343.       struct user_pass *up = NULL;
....
2416.               CLEAR(*up);
```

Use of Zero Initialized Pointer\Path 19:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=642
Status New

The variable declared in up at OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c in line 2335 is not initialized when it is used by up at OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c in line 2335.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Line	2343	2388
Object	up	up

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Method key_method_2_read(struct buffer *buf, struct tls_multi *multi, struct tls_session *session)

```
....
2343.       struct user_pass *up = NULL;
....
2388.       password_status = read_string(buf, up->password,
USER_PASS_LEN);
```

Use of Zero Initialized Pointer\Path 20:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=643
Status	New

The variable declared in up at OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c in line 2335 is not initialized when it is used by up at OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c in line 2335.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Line	2343	2387
Object	up	up

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Method key_method_2_read(struct buffer *buf, struct tls_multi *multi, struct tls_session *session)

```
....
2343.       struct user_pass *up = NULL;
....
2387.       username_status = read_string(buf, up->username,
USER_PASS_LEN);
```

Use of Zero Initialized Pointer\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=644
Status	New

The variable declared in ks_select at OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c in line 3907 is not initialized when it is used by save_ks at OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c in line 3934.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Line	3909	3950
Object	ks_select	save_ks

Code Snippet

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c

Method tls_select_encryption_key(struct tls_multi *multi)

```
....
3909.       struct key_state *ks_select = NULL;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c

Method tls_pre_encrypt(struct tls_multi *multi,

```
....
3950.           multi->save_ks = ks_select;
```

Use of Zero Initialized Pointer\Path 22:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=645
Status	New

The variable declared in prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788 is not initialized when it is used by prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	791	817
Object	prop_array	prop_array

Code Snippet

File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c

Method	BOOL DeletePartition(DWORD DriveIndex, ULONGLONG PartitionOffset, BOOL bSilent)

```
....
791.        VDS_PARTITION_PROP* prop_array = NULL;
....
817.                    hr =
IVdsAdvancedDisk_DeletePartition(pAdvancedDisk, prop_array[i].ullOffset,
TRUE, TRUE);
```

Use of Zero Initialized Pointer\Path 23:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=646
Status	New

The variable declared in prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788 is not initialized when it is used by prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	791	816
Object	prop_array	prop_array

Code Snippet

File Name	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method	BOOL DeletePartition(DWORD DriveIndex, ULONGLONG PartitionOffset, BOOL bSilent)

```
....
791.        VDS_PARTITION_PROP* prop_array = NULL;
....
816.                            prop_array[i].ullOffset,
SizeToHumanReadable(prop_array[i].ullSize, FALSE, FALSE));
```

Use of Zero Initialized Pointer\Path 24:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=647
Status	New

The variable declared in prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788 is not initialized when it is used by prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-	pbatard@@rufus-newest-CVE-2021-

	3520-FP.c	3520-FP.c
Line	791	816
Object	prop_array	prop_array

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL DeletePartition(DWORD DriveIndex, ULONGLONG PartitionOffset, BOOL bSilent)

```
....
791.        VDS_PARTITION_PROP* prop_array = NULL;
....
816.                            prop_array[i].ullOffset,
SizeToHumanReadable(prop_array[i].ullSize, FALSE, FALSE));
```

Use of Zero Initialized Pointer\Path 25:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=648
Status New

The variable declared in prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788 is not initialized when it is used by prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	791	816
Object	prop_array	prop_array

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL DeletePartition(DWORD DriveIndex, ULONGLONG PartitionOffset, BOOL bSilent)

```
....
791.        VDS_PARTITION_PROP* prop_array = NULL;
....
816.                            prop_array[i].ullOffset,
SizeToHumanReadable(prop_array[i].ullSize, FALSE, FALSE));
```

Use of Zero Initialized Pointer\Path 26:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=649
Status New

The variable declared in prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788 is not initialized when it is used by prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	791	815
Object	prop_array	prop_array

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL DeletePartition(DWORD DriveIndex, ULONGLONG PartitionOffset, BOOL bSilent)

```
....
791.        VDS_PARTITION_PROP* prop_array = NULL;
....
815.                    suprintf("● Partition %d (offset: %lld, size:
%s)", prop_array[i].ulPartitionNumber,
```

Use of Zero Initialized Pointer\Path 27:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=650
Status New

The variable declared in prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788 is not initialized when it is used by prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	791	816
Object	prop_array	prop_array

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL DeletePartition(DWORD DriveIndex, ULONGLONG PartitionOffset, BOOL bSilent)

```
....
791.        VDS_PARTITION_PROP* prop_array = NULL;
....
816.                            prop_array[i].ullOffset,
SizeToHumanReadable(prop_array[i].ullSize, FALSE, FALSE));
```

Use of Zero Initialized Pointer\Path 28:

Severity Medium

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=651
Status	New

The variable declared in prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788 is not initialized when it is used by prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	791	815
Object	prop_array	prop_array

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL DeletePartition(DWORD DriveIndex, ULONGLONG PartitionOffset, BOOL bSilent)

```
....
791.        VDS_PARTITION_PROP* prop_array = NULL;
....
815.                    suprintf("● Partition %d (offset: %lld, size:
%s)", prop_array[i].ulPartitionNumber,
```

Use of Zero Initialized Pointer\Path 29:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=652
Status	New

The variable declared in tick at openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c in line 262 is not initialized when it is used by ret at openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c in line 262.

	Source	Destination
File	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Line	377	399
Object	tick	ret

Code Snippet
File Name openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Method SSL_SESSION *d2i_SSL_SESSION_ex(SSL_SESSION **a, const unsigned char **pp,

```
....
377.           ret->ext.tick = NULL;
....
399.       OPENSSL_free(ret->ext.alpn_selected);
```

Use of Zero Initialized Pointer\Path 30:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=653
Status	New

The variable declared in Pointer at openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c in line 229 is not initialized when it is used by ret at openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c in line 262.

	Source	Destination
File	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Line	232	399
Object	Pointer	ret

Code Snippet

File Name openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c

Method static int ssl_session_strndup(char **pdst, ASN1_OCTET_STRING *src)

```
....
232.       *pdst = NULL;
```

File Name openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c

Method SSL_SESSION *d2i_SSL_SESSION_ex(SSL_SESSION **a, const unsigned char **pp,

```
....
399.       OPENSSL_free(ret->ext.alpn_selected);
```

Use of Zero Initialized Pointer\Path 31:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=654
Status	New

The variable declared in data at openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c in line 109 is not initialized when it is used by peer_rpk at openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c in line 109.

	Source	Destination
File	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Line	178	182
Object	data	peer_rpk

Code Snippet
File Name openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Method int i2d_SSL_SESSION(const SSL_SESSION *in, unsigned char **pp)

```
....
178.      peer_rpk.data = NULL;
....
182.              as.peer_rpk = &peer_rpk;
```

Use of Zero Initialized Pointer\Path 32:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=655
Status	New

The variable declared in Pointer at openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c in line 229 is not initialized when it is used by ret at openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c in line 262.

	Source	Destination
File	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Line	232	371
Object	Pointer	ret

Code Snippet
File Name openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Method static int ssl_session_strndup(char **pdst, ASN1_OCTET_STRING *src)

```
....
232.      *pdst = NULL;
```

File Name openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Method SSL_SESSION *d2i_SSL_SESSION_ex(SSL_SESSION **a, const unsigned char **pp,

```
....
371.      OPENSSL_free(ret->ext.tick);
```

Use of Zero Initialized Pointer\Path 33:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=656
Status	New

The variable declared in cipher at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 92 is not initialized when it is used by cipher at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 92.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	152	278
Object	cipher	cipher

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
152.              ctx->cipher = NULL;
....
278.          return ctx->cipher->einit(ctx->algctx,
```

Use of Zero Initialized Pointer\Path 34:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=657
Status	New

The variable declared in cipher at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 92 is not initialized when it is used by cipher at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 92.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	152	293
Object	cipher	cipher

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
152.              ctx->cipher = NULL;
....
293.      return ctx->cipher->dinit(ctx->algctx,
```

Use of Zero Initialized Pointer\Path 35:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=658
Status	New

The variable declared in cipher at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 92 is not initialized when it is used by cipher at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 92.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	152	220
Object	cipher	cipher

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
152.              ctx->cipher = NULL;
....
220.          ctx->algctx = ctx->cipher-
>newctx(ossl_provider_ctx(cipher->prov));
```

Use of Zero Initialized Pointer\Path 36:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=659
Status	New

The variable declared in cipher at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 92 is not initialized when it is used by cipher at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 92.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	152	218
Object	cipher	cipher

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
152.              ctx->cipher = NULL;
....
218.      ctx->cipher = cipher;
```

Use of Zero Initialized Pointer\Path 37:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=660
Status	New

The variable declared in fetched_cipher at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 92 is not initialized when it is used by cipher at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 92.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	154	218
Object	fetched_cipher	cipher

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
154.          ctx->fetched_cipher = NULL;
....
218.      ctx->cipher = cipher;
```

Use of Zero Initialized Pointer\Path 38:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=661
Status	New

The variable declared in cipher_data at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 92 is not initialized when it is used by cipher at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 92.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c

Line	165	218
Object	cipher_data	cipher

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
165.          ctx->cipher_data = NULL;
....
218.      ctx->cipher = cipher;
```

Use of Zero Initialized Pointer\Path 39:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=662
Status	New

The variable declared in tick at openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c in line 262 is not initialized when it is used by ret at openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c in line 262.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c
Line	377	399
Object	tick	ret

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c
Method SSL_SESSION *d2i_SSL_SESSION_ex(SSL_SESSION **a, const unsigned char **pp,

```
....
377.          ret->ext.tick = NULL;
....
399.      OPENSSL_free(ret->ext.alpn_selected);
```

Use of Zero Initialized Pointer\Path 40:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=663
Status	New

The variable declared in Pointer at openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c in line 229 is not initialized when it is used by ret at openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c in line 262.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c
Line	232	399
Object	Pointer	ret

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c
Method static int ssl_session_strndup(char **pdst, ASN1_OCTET_STRING *src)

```
....
232.        *pdst = NULL;
```

File Name openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c

Method SSL_SESSION *d2i_SSL_SESSION_ex(SSL_SESSION **a, const unsigned char **pp,

```
....
399.        OPENSSL_free(ret->ext.alpn_selected);
```

Use of Zero Initialized Pointer\Path 41:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=664
Status	New

The variable declared in data at openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c in line 109 is not initialized when it is used by peer_rpk at openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c in line 109.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c
Line	178	182
Object	data	peer_rpk

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c
Method int i2d_SSL_SESSION(const SSL_SESSION *in, unsigned char **pp)

```
....
178.        peer_rpk.data = NULL;
....
182.                as.peer_rpk = &peer_rpk;
```

Use of Zero Initialized Pointer\Path 42:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=665
Status	New

The variable declared in Pointer at openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c in line 229 is not initialized when it is used by ret at openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c in line 262.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c
Line	232	371
Object	Pointer	ret

Code Snippet

File Name openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c

Method static int ssl_session_strndup(char **pdst, ASN1_OCTET_STRING *src)

```
....
232.      *pdst = NULL;
```

File Name openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c

Method SSL_SESSION *d2i_SSL_SESSION_ex(SSL_SESSION **a, const unsigned char **pp,

```
....
371.      OPENSSL_free(ret->ext.tick);
```

Use of Zero Initialized Pointer\Path 43:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=666
Status	New

The variable declared in cipher at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 92 is not initialized when it is used by cipher at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 92.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	152	278
Object	cipher	cipher

Code Snippet

File Name	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method	static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
152.                ctx->cipher = NULL;
....
278.            return ctx->cipher->einit(ctx->algctx,
```

Use of Zero Initialized Pointer\Path 44:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=667
Status	New

The variable declared in cipher at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 92 is not initialized when it is used by cipher at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 92.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	152	293
Object	cipher	cipher

Code Snippet

File Name	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method	static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
152.                ctx->cipher = NULL;
....
293.        return ctx->cipher->dinit(ctx->algctx,
```

Use of Zero Initialized Pointer\Path 45:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=668
Status	New

The variable declared in cipher at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 92 is not initialized when it is used by cipher at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 92.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	152	220
Object	cipher	cipher

Code Snippet	
File Name	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method	static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
152.              ctx->cipher = NULL;
....
220.          ctx->algctx = ctx->cipher-
>newctx(ossl_provider_ctx(cipher->prov));
```

Use of Zero Initialized Pointer\Path 46:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=669
Status	New

The variable declared in cipher at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 92 is not initialized when it is used by cipher at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 92.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	152	218
Object	cipher	cipher

Code Snippet	
File Name	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method	static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
152.              ctx->cipher = NULL;
....
218.      ctx->cipher = cipher;
```

Use of Zero Initialized Pointer\Path 47:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=670
Status	New

The variable declared in fetched_cipher at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 92 is not initialized when it is used by cipher at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 92.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-	openssl@@openssl-openssl-3.3.1-CVE-

	2021-23840-FP.c	2021-23840-FP.c
Line	154	218
Object	fetched_cipher	cipher

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
154.          ctx->fetched_cipher = NULL;
....
218.      ctx->cipher = cipher;
```

Use of Zero Initialized Pointer\Path 48:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=671
Status	New

The variable declared in cipher_data at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 92 is not initialized when it is used by cipher at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 92.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	165	218
Object	cipher_data	cipher

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
165.          ctx->cipher_data = NULL;
....
218.      ctx->cipher = cipher;
```

Use of Zero Initialized Pointer\Path 49:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=672
Status	New

The variable declared in tick at openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c in line 262 is not initialized when it is used by ret at openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c in line 262.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c
Line	377	399
Object	tick	ret

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c
Method SSL_SESSION *d2i_SSL_SESSION_ex(SSL_SESSION **a, const unsigned char **pp,

```
....
377.           ret->ext.tick = NULL;
....
399.       OPENSSL_free(ret->ext.alpn_selected);
```

Use of Zero Initialized Pointer\Path 50:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=673
Status New

The variable declared in Pointer at openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c in line 229 is not initialized when it is used by ret at openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c in line 262.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c
Line	232	399
Object	Pointer	ret

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c
Method static int ssl_session_strndup(char **pdst, ASN1_OCTET_STRING *src)

```
....
232.       *pdst = NULL;
```

File Name openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c
Method SSL_SESSION *d2i_SSL_SESSION_ex(SSL_SESSION **a, const unsigned char **pp,

```
....
399.       OPENSSL_free(ret->ext.alpn_selected);
```

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=1020051&projectid=20044&pathid=555
Status	New

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	66	66
Object	i	i

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
66.       ret = asprintf(&(ef->file_entry[i].value), "%s\n%s", pre,
```

Memory Leak\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=556
Status	New

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	80	80
Object	i	i

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
80.     int ret = asprintf(&(ef->file_entry[i].comment_before_key),
```

Memory Leak\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=557
Status	New

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	106	106
Object	i	i

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
106.              int ret = asprintf(&(ef-
>file_entry[i].comment_after_value),
```

Memory Leak\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=558
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	66	66
Object	i	i

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
66.       ret = asprintf(&(ef->file_entry[i].value), "%s\n%s", pre,
```

Memory Leak\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20

	044&pathid=559
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	80	80
Object	i	i

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
80.     int ret = asprintf(&(ef->file_entry[i].comment_before_key),
```

Memory Leak\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=560
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	106	106
Object	i	i

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
106.              int ret = asprintf(&(ef-
>file_entry[i].comment_after_value),
```

Memory Leak\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=561
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.4.7-CVE-2023-	openSUSE@@libeconf-v0.4.7-CVE-2023-

	32181-TP.c	32181-TP.c
Line	112	112
Object	buffer	buffer

Code Snippet
File Name openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Method char *addbrackets(const char *string) {

```
....
112.      char *buffer = malloc(length + 3);
```

Memory Leak\Path 8:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=562
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Line	359	359
Object	out	out

Code Snippet
File Name OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Method management_callback_remote_entry_get(void *arg, unsigned int index, char **remote)

```
....
359.          char *out = malloc(len);
```

Memory Leak\Path 9:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=563
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c
Line	365	365
Object	out	out

Code Snippet	
File Name	OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c
Method	management_callback_remote_entry_get(void *arg, unsigned int index, char **remote)

```
....
365.            char *out = malloc(len);
```

Memory Leak\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=564
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c
Line	365	365
Object	out	out

Code Snippet	
File Name	OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c
Method	management_callback_remote_entry_get(void *arg, unsigned int index, char **remote)

```
....
365.            char *out = malloc(len);
```

Memory Leak\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=565
Status	New

	Source	Destination
File	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	215	215
Object	group	group

Code Snippet	
File Name	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method	struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
215.    copied_fe.group = strdup(fe.group);
```

Memory Leak\Path 12:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=566
Status	New

	Source	Destination
File	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	216	216
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
216.    copied_fe.key = strdup(fe.key);
```

Memory Leak\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=567
Status	New

	Source	Destination
File	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	218	218
Object	value	value

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
218.      copied_fe.value = strdup(fe.value);
```

Memory Leak\Path 14:

Severity	Medium

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=568
Status	New

	Source	Destination
File	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	222	222
Object	comment_before_key	comment_before_key

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
222.      copied_fe.comment_before_key = strdup(fe.comment_before_key);
```

Memory Leak\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=569
Status	New

	Source	Destination
File	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	226	226
Object	comment_after_value	comment_after_value

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
226.      copied_fe.comment_after_value =
strdup(fe.comment_after_value);
```

Memory Leak\Path 16:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=570
Status	New

	Source	Destination
File	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	45	45
Object	group	group

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
45.     key_file->file_entry[num].group = strdup(KEY_FILE_NULL_VALUE);
```

Memory Leak\Path 17:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=571
Status New

	Source	Destination
File	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	46	46
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
46.     key_file->file_entry[num].key = strdup(KEY_FILE_NULL_VALUE);
```

Memory Leak\Path 18:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=572
Status New

	Source	Destination
File	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	47	47

Object	value	value

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
47.    key_file->file_entry[num].value = strdup(KEY_FILE_NULL_VALUE);
```

Memory Leak\Path 19:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=573
Status New

	Source	Destination
File	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	83	83
Object	absolute_path	absolute_path

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method char *get_absolute_path(const char *path, econf_err *error) {

```
....
83.      absolute_path = strdup(buffer);
```

Memory Leak\Path 20:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=574
Status New

	Source	Destination
File	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	85	85
Object	absolute_path	absolute_path

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method char *get_absolute_path(const char *path, econf_err *error) {

```
....
85.      absolute_path = strdup(path);
```

Memory Leak\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=575
Status	New

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	56	56
Object	value	value

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
56.      ef->file_entry[i].value = strdup("");
```

Memory Leak\Path 22:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=576
Status	New

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	104	104
Object	comment_after_value	comment_after_value

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
104.              ef->file_entry[i].comment_after_value = strdup(post);
```

Memory Leak\Path 23:

Severity	Medium

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=577
Status	New

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	191	191
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method store (econf_file *ef, const char *group, const char *key,

```
....
191.      ef->file_entry[ef->length-1].key = strdup(key);
```

Memory Leak\Path 24:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=578
Status	New

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	193	193
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method store (econf_file *ef, const char *group, const char *key,

```
....
193.      ef->file_entry[ef->length-1].key =
strdup(KEY_FILE_NULL_VALUE);
```

Memory Leak\Path 25:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=579
Status	New

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	254	254
Object	path	path

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
254.    ef->path = strdup (file);
```

Memory Leak\Path 26:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=580
Status New

	Source	Destination
File	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	211	211
Object	group	group

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
211.    copied_fe.group = strdup(fe.group);
```

Memory Leak\Path 27:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=581
Status New

	Source	Destination
File	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	212	212

Object	key	key

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
212.    copied_fe.key = strdup(fe.key);
```

Memory Leak\Path 28:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=582
Status New

	Source	Destination
File	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	213	213
Object	value	value

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
213.    copied_fe.value = strdup(fe.value);
```

Memory Leak\Path 29:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=583
Status New

	Source	Destination
File	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	45	45
Object	group	group

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
45.    key_file->file_entry[num].group = strdup(KEY_FILE_NULL_VALUE);
```

Memory Leak\Path 30:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=584
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	46	46
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
46.    key_file->file_entry[num].key = strdup(KEY_FILE_NULL_VALUE);
```

Memory Leak\Path 31:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=585
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	47	47
Object	value	value

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
47.    key_file->file_entry[num].value = strdup(KEY_FILE_NULL_VALUE);
```

Memory Leak\Path 32:

Severity	Medium

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=586
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	81	81
Object	absolute_path	absolute_path

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method char *get_absolute_path(const char *path, econf_err *error) {

```
....
81.       absolute_path = strdup(buffer);
```

Memory Leak\Path 33:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=587
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	83	83
Object	absolute_path	absolute_path

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method char *get_absolute_path(const char *path, econf_err *error) {

```
....
83.       absolute_path = strdup(path);
```

Memory Leak\Path 34:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=588
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	211	211
Object	group	group

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
211.    copied_fe.group = strdup(fe.group);
```

Memory Leak\Path 35:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=589
Status New

	Source	Destination
File	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	212	212
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
212.    copied_fe.key = strdup(fe.key);
```

Memory Leak\Path 36:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=590
Status New

	Source	Destination
File	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	214	214

Object	value	value

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
214.      copied_fe.value = strdup(fe.value);
```

Memory Leak\Path 37:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=591
Status New

	Source	Destination
File	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	45	45
Object	group	group

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
45.    key_file->file_entry[num].group = strdup(KEY_FILE_NULL_VALUE);
```

Memory Leak\Path 38:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=592
Status New

	Source	Destination
File	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	46	46
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
46.    key_file->file_entry[num].key = strdup(KEY_FILE_NULL_VALUE);
```

Memory Leak\Path 39:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=593
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	47	47
Object	value	value

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
47.    key_file->file_entry[num].value = strdup(KEY_FILE_NULL_VALUE);
```

Memory Leak\Path 40:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=594
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	81	81
Object	absolute_path	absolute_path

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method char *get_absolute_path(const char *path, econf_err *error) {

```
....
81.      absolute_path = strdup(buffer);
```

Memory Leak\Path 41:

Severity	Medium

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=595
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	83	83
Object	absolute_path	absolute_path

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method char *get_absolute_path(const char *path, econf_err *error) {

```
....
83.      absolute_path = strdup(path);
```

Memory Leak\Path 42:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=596
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	215	215
Object	group	group

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
215.    copied_fe.group = strdup(fe.group);
```

Memory Leak\Path 43:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=597
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	216	216
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
216.    copied_fe.key = strdup(fe.key);
```

Memory Leak\Path 44:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=598
Status New

	Source	Destination
File	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	218	218
Object	value	value

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
218.      copied_fe.value = strdup(fe.value);
```

Memory Leak\Path 45:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=599
Status New

	Source	Destination
File	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	222	222

Object	comment_before_key	comment_before_key

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
222.      copied_fe.comment_before_key = strdup(fe.comment_before_key);
```

Memory Leak\Path 46:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=600
Status New

	Source	Destination
File	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	226	226
Object	comment_after_value	comment_after_value

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
226.      copied_fe.comment_after_value =
strdup(fe.comment_after_value);
```

Memory Leak\Path 47:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=601
Status New

	Source	Destination
File	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	45	45
Object	group	group

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
45.    key_file->file_entry[num].group = strdup(KEY_FILE_NULL_VALUE);
```

Memory Leak\Path 48:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=602
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	46	46
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
46.    key_file->file_entry[num].key = strdup(KEY_FILE_NULL_VALUE);
```

Memory Leak\Path 49:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=603
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	47	47
Object	value	value

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
47.    key_file->file_entry[num].value = strdup(KEY_FILE_NULL_VALUE);
```

Memory Leak\Path 50:

Severity	Medium

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=604
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	83	83
Object	absolute_path	absolute_path

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method char *get_absolute_path(const char *path, econf_err *error) {

```
....
83.      absolute_path = strdup(buffer);
```

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=1020051&projectid=20044&pathid=66
Status	New

The size of the buffer used by evp_cipher_init_internal in q, at line 92 of openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that evp_cipher_init_internal passes to q, at line 92 of openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c, to overwrite the target buffer.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	253	253
Object	q	q

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
253.              memcpy(q++, p, sizeof(*q));
```

Buffer Overflow boundcpy WrongSizeParam\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=67
Status	New

The size of the buffer used by evp_cipher_init_internal in q, at line 92 of openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that evp_cipher_init_internal passes to q, at line 92 of openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c, to overwrite the target buffer.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	261	261
Object	q	q

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
261.              memcpy(q++, p, sizeof(*q));
```

Buffer Overflow boundcpy WrongSizeParam\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=68
Status	New

The size of the buffer used by evp_cipher_init_internal in q, at line 92 of openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that evp_cipher_init_internal passes to q, at line 92 of openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c, to overwrite the target buffer.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	253	253
Object	q	q

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c

Method	static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
253.              memcpy(q++, p, sizeof(*q));
```

Buffer Overflow boundcpy WrongSizeParam\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=69
Status	New

The size of the buffer used by evp_cipher_init_internal in q, at line 92 of openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that evp_cipher_init_internal passes to q, at line 92 of openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c, to overwrite the target buffer.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	261	261
Object	q	q

Code Snippet	
File Name	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method	static int evp_cipher_init_internal(EVP_CIPHER_CTX *ctx,

```
....
261.              memcpy(q++, p, sizeof(*q));
```

Buffer Overflow boundcpy WrongSizeParam\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=70
Status	New

The size of the buffer used by do_init_crypto_tls in Namespace1365619970, at line 3221 of OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that do_init_crypto_tls passes to Namespace1365619970, at line 3221 of OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Line	3325	3325
Object	Namespace1365619970	Namespace1365619970

Code Snippet
File Name OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Method do_init_crypto_tls(struct context *c, const unsigned int flags)

```
....
3325.       memcpy(to.remote_cert_ku, options->remote_cert_ku,
sizeof(to.remote_cert_ku));
```

Buffer Overflow boundcpy WrongSizeParam\Path 6:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=71
Status New

The size of the buffer used by do_init_crypto_tls in Namespace1365619970, at line 3221 of OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that do_init_crypto_tls passes to Namespace1365619970, at line 3221 of OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Line	3332	3332
Object	Namespace1365619970	Namespace1365619970

Code Snippet
File Name OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Method do_init_crypto_tls(struct context *c, const unsigned int flags)

```
....
3332.       memcpy(to.x509_username_field, options->x509_username_field,
sizeof(to.x509_username_field));
```

Buffer Overflow boundcpy WrongSizeParam\Path 7:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=72
Status New

The size of the buffer used by do_init_crypto_tls in Namespace1706540875, at line 3238 of OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that do_init_crypto_tls passes to Namespace1706540875, at line 3238 of OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.7-CVE-2023-	OpenVPN@@openvpn-v2.6.7-CVE-2023-

	46849-FP.c	46849-FP.c
Line	3343	3343
Object	Namespace1706540875	Namespace1706540875

Code Snippet
File Name OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c
Method do_init_crypto_tls(struct context *c, const unsigned int flags)

```
....
3343.       memcpy(to.remote_cert_ku, options->remote_cert_ku,
sizeof(to.remote_cert_ku));
```

Buffer Overflow boundcpy WrongSizeParam\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=73
Status	New

The size of the buffer used by do_init_crypto_tls in Namespace1706540875, at line 3238 of OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that do_init_crypto_tls passes to Namespace1706540875, at line 3238 of OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c
Line	3350	3350
Object	Namespace1706540875	Namespace1706540875

Code Snippet
File Name OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c
Method do_init_crypto_tls(struct context *c, const unsigned int flags)

```
....
3350.       memcpy(to.x509_username_field, options->x509_username_field,
sizeof(to.x509_username_field));
```

Buffer Overflow boundcpy WrongSizeParam\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=74
Status	New

The size of the buffer used by do_init_crypto_tls in Namespace1043124911, at line 3227 of OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that do_init_crypto_tls passes to

Namespace1043124911, at line 3227 of OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c
Line	3331	3331
Object	Namespace1043124911	Namespace1043124911

Code Snippet
File Name OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c
Method do_init_crypto_tls(struct context *c, const unsigned int flags)

```
....
3331.       memcpy(to.remote_cert_ku, options->remote_cert_ku,
sizeof(to.remote_cert_ku));
```

Buffer Overflow boundcpy WrongSizeParam\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=75
Status	New

The size of the buffer used by do_init_crypto_tls in Namespace1043124911, at line 3227 of OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that do_init_crypto_tls passes to Namespace1043124911, at line 3227 of OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c
Line	3338	3338
Object	Namespace1043124911	Namespace1043124911

Code Snippet
File Name OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c
Method do_init_crypto_tls(struct context *c, const unsigned int flags)

```
....
3338.       memcpy(to.x509_username_field, options->x509_username_field,
sizeof(to.x509_username_field));
```

Buffer Overflow boundcpy WrongSizeParam\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=76

Status	New

The size of the buffer used by do_init_crypto_tls in Namespace401786297, at line 2663 of OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that do_init_crypto_tls passes to Namespace401786297, at line 2663 of OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c
Line	2777	2777
Object	Namespace401786297	Namespace401786297

Code Snippet
File Name OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c
Method do_init_crypto_tls(struct context *c, const unsigned int flags)

```
....
2777.       memmove(to.remote_cert_ku, options->remote_cert_ku,
sizeof(to.remote_cert_ku));
```

Buffer Overflow boundcpy WrongSizeParam\Path 12:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=77
Status	New

The size of the buffer used by GetDrivePartitionData in Namespace981654722, at line 1851 of pbatard@@rufus-newest-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 GetDrivePartitionData passes to Namespace981654722, at line 1851 of pbatard@@rufus-newest-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	1867	1867
Object	Namespace981654722	Namespace981654722

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL GetDrivePartitionData(DWORD DriveIndex, char* FileSystemName, DWORD FileSystemNameSize, BOOL bSilent)

```
....
1867.       memset(SelectedDrive.Partition, 0,
sizeof(SelectedDrive.Partition));
```

Buffer Overflow boundcpy WrongSizeParam\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=78
Status	New

The size of the buffer used by CreatePartition in Namespace981654722, at line 2254 of pbatard@@rufus-newest-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 CreatePartition passes to Namespace981654722, at line 2254 of pbatard@@rufus-newest-CVE-2021-3520-FP.c, to overwrite the target buffer.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	2289	2289
Object	Namespace981654722	Namespace981654722

Code Snippet

File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c

Method BOOL CreatePartition(HANDLE hDrive, int partition_style, int file_system, BOOL mbr_uefi_marker, uint8_t extra_partitions)

```
....
2289.       memset(SelectedDrive.Partition, 0,
sizeof(SelectedDrive.Partition));
```

Buffer Overflow boundcpy WrongSizeParam\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=79
Status	New

The size of the buffer used by *codes_join in a, at line 165 of pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *codes_join passes to a, at line 165 of pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c, to overwrite the target buffer.

	Source	Destination
File	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Line	168	168
Object	a	a

Code Snippet

File Name pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c

Method static Jsi_OpCodes *codes_join(Jsi_OpCodes *a, Jsi_OpCodes *b)

```
....
168.      memcpy(ret->codes, a->codes, a->code_len *
sizeof(jsi_OpCode));
```

Buffer Overflow boundcpy WrongSizeParam\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=80
Status	New

The size of the buffer used by *codes_join in jsi_OpCode, at line 165 of pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *codes_join passes to jsi_OpCode, at line 165 of pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c, to overwrite the target buffer.

	Source	Destination
File	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Line	168	168
Object	jsi_OpCode	jsi_OpCode

Code Snippet
File Name pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Method static Jsi_OpCodes *codes_join(Jsi_OpCodes *a, Jsi_OpCodes *b)

```
....
168.      memcpy(ret->codes, a->codes, a->code_len *
sizeof(jsi_OpCode));
```

Buffer Overflow boundcpy WrongSizeParam\Path 16:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=81
Status	New

The size of the buffer used by *codes_join in b, at line 165 of pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *codes_join passes to b, at line 165 of pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c, to overwrite the target buffer.

	Source	Destination
File	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Line	169	169
Object	b	b

Code Snippet
File Name pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Method static Jsi_OpCodes *codes_join(Jsi_OpCodes *a, Jsi_OpCodes *b)

```
....
169.      memcpy(&ret->codes[a->code_len], b->codes, b->code_len *
sizeof(jsi_OpCode));
```

Buffer Overflow boundcpy WrongSizeParam\Path 17:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=82
Status New

The size of the buffer used by *codes_join in jsi_OpCode, at line 165 of pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *codes_join passes to jsi_OpCode, at line 165 of pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c, to overwrite the target buffer.

	Source	Destination
File	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Line	169	169
Object	jsi_OpCode	jsi_OpCode

Code Snippet
File Name pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Method static Jsi_OpCodes *codes_join(Jsi_OpCodes *a, Jsi_OpCodes *b)

```
....
169.      memcpy(&ret->codes[a->code_len], b->codes, b->code_len *
sizeof(jsi_OpCode));
```

Buffer Overflow boundcpy WrongSizeParam\Path 18:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=83
Status New

The size of the buffer used by ssl_session_memcpy in src, at line 243 of openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ssl_session_memcpy passes to src, at line 243 of openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c, to overwrite the target buffer.

	Source	Destination
File	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Line	252	252

Object	src	src

Code Snippet
File Name openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Method static int ssl_session_memcpy(unsigned char *dst, size_t *pdstlen,

```
....
252.      memcpy(dst, src->data, src->length);
```

Buffer Overflow boundcpy WrongSizeParam\Path 19:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=84
Status	New

The size of the buffer used by EVP_CIPHER_CTX_copy in in, at line 1458 of openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that EVP_CIPHER_CTX_copy passes to in, at line 1458 of openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c, to overwrite the target buffer.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	1511	1511
Object	in	in

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method int EVP_CIPHER_CTX_copy(EVP_CIPHER_CTX *out, const EVP_CIPHER_CTX *in)

```
....
1511.          memcpy(out->cipher_data, in->cipher_data, in->cipher-
>ctx_size);
```

Buffer Overflow boundcpy WrongSizeParam\Path 20:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=85
Status	New

The size of the buffer used by extension_append in extension_length, at line 746 of openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that extension_append passes to extension_length, at line 746 of openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c, to overwrite the target buffer.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-	openssl@@openssl-openssl-3.2.1-CVE-

	2021-3449-FP.c	2021-3449-FP.c
Line	761	761
Object	extension_length	extension_length

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c
Method static void extension_append(unsigned int version,

```
....
761.      memcpy(serverinfo + contextoff, extension, extension_length);
```

Buffer Overflow boundcpy WrongSizeParam\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=86
Status	New

The size of the buffer used by SSL_CTX_use_serverinfo_ex in serverinfo_length, at line 764 of openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that SSL_CTX_use_serverinfo_ex passes to serverinfo_length, at line 764 of openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c, to overwrite the target buffer.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c
Line	810	810
Object	serverinfo_length	serverinfo_length

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c
Method int SSL_CTX_use_serverinfo_ex(SSL_CTX *ctx, unsigned int version,

```
....
810.      memcpy(ctx->cert->key->serverinfo, serverinfo,
serverinfo_length);
```

Buffer Overflow boundcpy WrongSizeParam\Path 22:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=87
Status	New

The size of the buffer used by ssl_session_memcpy in src, at line 243 of openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer

overflow attack, using the source buffer that ssl_session_memcpy passes to src, at line 243 of openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c, to overwrite the target buffer.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c
Line	252	252
Object	src	src

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c
Method static int ssl_session_memcpy(unsigned char *dst, size_t *pdstlen,

```
....
252.      memcpy(dst, src->data, src->length);
```

Buffer Overflow boundcpy WrongSizeParam\Path 23:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=88
Status New

The size of the buffer used by EVP_CIPHER_CTX_copy in in, at line 1458 of openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that EVP_CIPHER_CTX_copy passes to in, at line 1458 of openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c, to overwrite the target buffer.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	1511	1511
Object	in	in

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method int EVP_CIPHER_CTX_copy(EVP_CIPHER_CTX *out, const EVP_CIPHER_CTX *in)

```
....
1511.          memcpy(out->cipher_data, in->cipher_data, in->cipher-
>ctx_size);
```

Buffer Overflow boundcpy WrongSizeParam\Path 24:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=89
Status New

The size of the buffer used by extension_append in extension_length, at line 746 of openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that extension_append passes to extension_length, at line 746 of openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c, to overwrite the target buffer.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c
Line	761	761
Object	extension_length	extension_length

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c
Method static void extension_append(unsigned int version,

```
....
761.      memcpy(serverinfo + contextoff, extension, extension_length);
```

Buffer Overflow boundcpy WrongSizeParam\Path 25:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=90
Status New

The size of the buffer used by SSL_CTX_use_serverinfo_ex in serverinfo_length, at line 764 of openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that SSL_CTX_use_serverinfo_ex passes to serverinfo_length, at line 764 of openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c, to overwrite the target buffer.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c
Line	810	810
Object	serverinfo_length	serverinfo_length

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c
Method int SSL_CTX_use_serverinfo_ex(SSL_CTX *ctx, unsigned int version,

```
....
810.      memcpy(ctx->cert->key->serverinfo, serverinfo,
serverinfo_length);
```

Buffer Overflow boundcpy WrongSizeParam\Path 26:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20

	044&pathid=91
Status	New

The size of the buffer used by ssl_session_memcpy in src, at line 243 of openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ssl_session_memcpy passes to src, at line 243 of openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c, to overwrite the target buffer.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c
Line	252	252
Object	src	src

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c
Method static int ssl_session_memcpy(unsigned char *dst, size_t *pdstlen,

```
....
252.      memcpy(dst, src->data, src->length);
```

Buffer Overflow boundcpy WrongSizeParam\Path 27:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=92
Status	New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1330 of OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1330 of OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Line	1447	1447
Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1447.          memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 28:

Severity	Medium

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=93
Status	New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1330 of OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1330 of OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Line	1448	1448
Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1448.          memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 29:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=94
Status	New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1330 of OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1330 of OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c
Line	1447	1447
Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1447.           memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 30:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=95
Status	New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1330 of OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1330 of OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c
Line	1448	1448
Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1448.           memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 31:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=96
Status	New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Line	1449	1449

Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1449.           memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 32:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=97
Status	New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Line	1450	1450
Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1450.           memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 33:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=98
Status	New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Line	1449	1449
Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1449.           memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 34:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=99
Status New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Line	1450	1450
Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1450.           memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 35:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=100
Status New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Line	1449	1449
Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1449.           memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 36:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=101
Status New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Line	1450	1450
Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1450.           memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 37:

Severity Medium
Result State To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=102
Status	New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Line	1449	1449
Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1449.           memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 38:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=103
Status	New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Line	1450	1450
Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1450.          memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 39:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=104
Status	New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Line	1449	1449
Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1449.          memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 40:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=105
Status	New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1332 of OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Line	1450	1450

Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1450.           memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 41:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=106
Status	New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1493 of OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1493 of OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	1610	1610
Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1610.           memcpy(ethhdr.source, orig_ethhdr->dest,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 42:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=107
Status	New

The size of the buffer used by ipv6_send_icmp_unreachable in OPENVPN_ETH_ALEN, at line 1493 of OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ipv6_send_icmp_unreachable passes to OPENVPN_ETH_ALEN, at line 1493 of OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	1611	1611
Object	OPENVPN_ETH_ALEN	OPENVPN_ETH_ALEN

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method ipv6_send_icmp_unreachable(struct context *c, struct buffer *buf, bool client)

```
....
1611.           memcpy(ethhdr.dest, orig_ethhdr->source,
OPENVPN_ETH_ALEN);
```

Buffer Overflow boundcpy WrongSizeParam\Path 43:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=108
Status New

The size of the buffer used by key_ctx_update_implicit_iv in impl_iv_len, at line 1678 of OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that key_ctx_update_implicit_iv passes to impl_iv_len, at line 1678 of OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c, to overwrite the target buffer.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Line	1688	1688
Object	impl_iv_len	impl_iv_len

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Method key_ctx_update_implicit_iv(struct key_ctx *ctx, uint8_t *key, size_t key_len)

```
....
1688.           memcpy(ctx->implicit_iv, key, impl_iv_len);
```

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=1020051&projectid=20

	044&pathid=109
Status	New

Calling free() (line 41) on a variable that was not dynamically allocated (line 41) in file openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c may result with a crash.

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	70	70
Object	pre	pre

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
70.         free(pre);
```

MemoryFree on StackVariable\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=110
Status	New

Calling free() (line 41) on a variable that was not dynamically allocated (line 41) in file openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c may result with a crash.

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	84	84
Object	pre	pre

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
84.       free(pre);
```

MemoryFree on StackVariable\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=111

Status	New

Calling free() (line 41) on a variable that was not dynamically allocated (line 41) in file openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c may result with a crash.

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	110	110
Object	pre	pre

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
110.                free(pre);
```

MemoryFree on StackVariable\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=112
Status	New

Calling free() (line 41) on a variable that was not dynamically allocated (line 41) in file openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c may result with a crash.

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	70	70
Object	pre	pre

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
70.         free(pre);
```

MemoryFree on StackVariable\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=113
Status	New

Calling free() (line 41) on a variable that was not dynamically allocated (line 41) in file openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c may result with a crash.

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	84	84
Object	pre	pre

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
84.     free(pre);
```

MemoryFree on StackVariable\Path 6:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=114
Status New

Calling free() (line 41) on a variable that was not dynamically allocated (line 41) in file openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c may result with a crash.

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	110	110
Object	pre	pre

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
110.                free(pre);
```

MemoryFree on StackVariable\Path 7:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=115
Status New

Calling free() (line 663) on a variable that was not dynamically allocated (line 663) in file OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c may result with a crash.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Line	740	740
Object	cert	cert

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Method init_ssl(const struct options *options, struct tls_root_ctx *new_ctx, bool in_chroot)

```
....
740.            free(cert);
```

MemoryFree on StackVariable\Path 8:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=116
Status New

Calling free() (line 993) on a variable that was not dynamically allocated (line 993) in file pbatard@@rufus-newest-CVE-2021-3520-FP.c may result with a crash.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	1005	1005
Object	LogicalPath	LogicalPath

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL WaitForLogical(DWORD DriveIndex, uint64_t PartitionOffset)

```
....
1005.                   free(LogicalPath);
```

MemoryFree on StackVariable\Path 9:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=117
Status New

Calling free() (line 993) on a variable that was not dynamically allocated (line 993) in file pbatard@@rufus-newest-CVE-2021-3520-FP.c may result with a crash.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	1008	1008
Object	LogicalPath	LogicalPath

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL WaitForLogical(DWORD DriveIndex, uint64_t PartitionOffset)

```
....
1008.              free(LogicalPath);
```

MemoryFree on StackVariable\Path 10:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=118
Status New

Calling free() (line 1022) on a variable that was not dynamically allocated (line 1022) in file pbatard@@rufus-newest-CVE-2021-3520-FP.c may result with a crash.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	1033	1033
Object	LogicalPath	LogicalPath

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method HANDLE GetLogicalHandle(DWORD DriveIndex, uint64_t PartitionOffset, BOOL bLockDrive, BOOL bWriteAccess, BOOL bWriteShare)

```
....
1033.        free(LogicalPath);
```

MemoryFree on StackVariable\Path 11:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=119
Status New

Calling free() (line 1038) on a variable that was not dynamically allocated (line 1038) in file pbatard@@rufus-newest-CVE-2021-3520-FP.c may result with a crash.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	1049	1049
Object	LogicalPath	LogicalPath

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method HANDLE AltGetLogicalHandle(DWORD DriveIndex, uint64_t PartitionOffset, BOOL bLockDrive, BOOL bWriteAccess, BOOL bWriteShare)

```
....
1049.       free(LogicalPath);
```

MemoryFree on StackVariable\Path 12:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=120
Status New

Calling free() (line 1568) on a variable that was not dynamically allocated (line 1568) in file pbatard@@rufus-newest-CVE-2021-3520-FP.c may result with a crash.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	1706	1706
Object	volume_name	volume_name

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL ToggleEsp(DWORD DriveIndex, uint64_t PartitionOffset)

```
....
1706.                   free(volume_name);
```

MemoryFree on StackVariable\Path 13:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=121
Status New

Calling free() (line 2153) on a variable that was not dynamically allocated (line 2153) in file pbatard@@rufus-newest-CVE-2021-3520-FP.c may result with a crash.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	2179	2179
Object	volume_name	volume_name

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method char* AltMountVolume(DWORD DriveIndex, uint64_t PartitionOffset, BOOL bSilent)

```
....
2179.        free(volume_name);
```

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=1020051&projectid=20044&pathid=123
Status New

The function combined_len in openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c at line 35 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	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	38	38
Object	combined_len	combined_len

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method char *combine_strings(const char *string_one, const char *string_two,

```
....
38.     char *combined = malloc(combined_len);
```

Wrong Size t Allocation\Path 2:

Severity Medium
Result State To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=124
Status	New

The function combined_len in openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c at line 35 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	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	38	38
Object	combined_len	combined_len

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method char *combine_strings(const char *string_one, const char *string_two,

```
....
38.    char *combined = malloc(combined_len);
```

Wrong Size t Allocation\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=125
Status	New

The function combined_len in openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c at line 35 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	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	38	38
Object	combined_len	combined_len

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method char *combine_strings(const char *string_one, const char *string_two,

```
....
38.    char *combined = malloc(combined_len);
```

Wrong Size t Allocation\Path 4:

Severity	Medium

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=126
Status	New

The function combined_len in openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c at line 35 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	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	38	38
Object	combined_len	combined_len

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method char *combine_strings(const char *string_one, const char *string_two,

```
....
38.    char *combined = malloc(combined_len);
```

Wrong Size t Allocation\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=127
Status	New

The function combined_len in openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c at line 35 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	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Line	38	38
Object	combined_len	combined_len

Code Snippet
File Name openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Method char *combine_strings(const char *string_one, const char *string_two,

```
....
38.    char *combined = malloc(combined_len);
```

Wrong Size t Allocation\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=128
Status	New

The function sector_size in pbatard@@rufus-newest-CVE-2021-3520-FP.c at line 1719 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	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	1747	1747
Object	sector_size	sector_size

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method const char* GetFsName(HANDLE hPhysical, LARGE_INTEGER StartingOffset)

```
....
1747.       uint8_t* buf = calloc(sector_size, 1);
```

Wrong Size t Allocation\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=129
Status	New

The function size in pbatard@@rufus-newest-CVE-2021-3520-FP.c at line 2227 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	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	2230	2230
Object	size	size

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method static BOOL ClearPartition(HANDLE hDrive, uint64_t offset, DWORD size)

```
....
2230.       uint8_t* buffer = calloc(size, 1);
```

Wrong Size t Allocation\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=130
Status	New

The function length in openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c at line 105 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	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	108	108
Object	length	length

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method char *addbrackets(const char *string) {

```
....
108.        char *buffer = malloc(length + 3);
```

Wrong Size t Allocation\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=131
Status	New

The function length in openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c at line 103 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	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	106	106
Object	length	length

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method char *addbrackets(const char *string) {

```
....
106.        char *buffer = malloc(length + 3);
```

Wrong Size t Allocation\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=132
Status	New

The function length in openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c at line 103 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	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	106	106
Object	length	length

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method char *addbrackets(const char *string) {

```
....
106.      char *buffer = malloc(length + 3);
```

Wrong Size t Allocation\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=133
Status	New

The function length in openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c at line 105 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	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	108	108
Object	length	length

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method char *addbrackets(const char *string) {

```
....
108.      char *buffer = malloc(length + 3);
```

Wrong Size t Allocation\Path 12:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=134
Status	New

The function length in openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c at line 107 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	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Line	112	112
Object	length	length

Code Snippet
File Name openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Method char *addbrackets(const char *string) {

```
....
112.      char *buffer = malloc(length + 3);
```

Char Overflow

Query Path:
CPP\Cx\CPP Integer Overflow\Char Overflow 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)

Description

Char Overflow\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=344
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 109 of openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination

File	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Line	150	150
Object	AssignExpr	AssignExpr

Code Snippet
File Name openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Method int i2d_SSL_SESSION(const SSL_SESSION *in, unsigned char **pp)

```
....
150.      cipher_data[0] = ((unsigned char)(l >> 8L)) & 0xff;
```

Char Overflow\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=345
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 109 of openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Line	151	151
Object	AssignExpr	AssignExpr

Code Snippet
File Name openssl@@openssl-openssl-3.2.0-alpha1-CVE-2023-0216-FP.c
Method int i2d_SSL_SESSION(const SSL_SESSION *in, unsigned char **pp)

```
....
151.      cipher_data[1] = ((unsigned char)(l)) & 0xff;
```

Char Overflow\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=346
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 109 of openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination

File	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c
Line	150	150
Object	AssignExpr	AssignExpr

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c
Method int i2d_SSL_SESSION(const SSL_SESSION *in, unsigned char **pp)

```
....
150.      cipher_data[0] = ((unsigned char)(l >> 8L)) & 0xff;
```

Char Overflow\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=347
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 109 of openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c
Line	151	151
Object	AssignExpr	AssignExpr

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2023-0216-FP.c
Method int i2d_SSL_SESSION(const SSL_SESSION *in, unsigned char **pp)

```
....
151.      cipher_data[1] = ((unsigned char)(l)) & 0xff;
```

Char Overflow\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=348
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 109 of openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination

File	openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c
Line	150	150
Object	AssignExpr	AssignExpr

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c
Method int i2d_SSL_SESSION(const SSL_SESSION *in, unsigned char **pp)

```
....
150.      cipher_data[0] = ((unsigned char)(l >> 8L)) & 0xff;
```

Char Overflow\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=349
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 109 of openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c
Line	151	151
Object	AssignExpr	AssignExpr

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2023-0216-FP.c
Method int i2d_SSL_SESSION(const SSL_SESSION *in, unsigned char **pp)

```
....
151.      cipher_data[1] = ((unsigned char)(l)) & 0xff;
```

Integer Overflow

Query Path:
CPP\Cx\CPP Integer Overflow\Integer Overflow Version:0

Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.2 - Buffer overflows
FISMA 2014: System And Information Integrity
NIST SP 800-53: SI-10 Information Input Validation (P1)

Description

Integer Overflow\Path 1:

Severity	Medium
Result State	To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=350
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 2254 of pbatard@@rufus-newest-CVE-2021-3520-FP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	2316	2316
Object	AssignExpr	AssignExpr

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL CreatePartition(HANDLE hDrive, int partition_style, int file_system, BOOL mbr_uefi_marker, uint8_t extra_partitions)

```
....
2316.             partition_index[PI_ESP] = pi;
```

Integer Overflow\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=351
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 2254 of pbatard@@rufus-newest-CVE-2021-3520-FP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	2355	2355
Object	AssignExpr	AssignExpr

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL CreatePartition(HANDLE hDrive, int partition_style, int file_system, BOOL mbr_uefi_marker, uint8_t extra_partitions)

```
....
2355.                   partition_index[PI_CASPER] = pi;
```

Integer Overflow\Path 3:

Severity	Medium

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=352
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 2254 of pbatard@@rufus-newest-CVE-2021-3520-FP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	2360	2360
Object	AssignExpr	AssignExpr

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL CreatePartition(HANDLE hDrive, int partition_style, int file_system, BOOL mbr_uefi_marker, uint8_t extra_partitions)

```
....
2360.                    partition_index[PI_ESP] = pi;
```

Integer Overflow\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=353
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 2254 of pbatard@@rufus-newest-CVE-2021-3520-FP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	2364	2364
Object	AssignExpr	AssignExpr

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL CreatePartition(HANDLE hDrive, int partition_style, int file_system, BOOL mbr_uefi_marker, uint8_t extra_partitions)

```
....
2364.                    partition_index[PI_UEFI_NTFS] = pi;
```

Heap Inspection

Query Path:
CPP\Cx\CPP Medium Threat\Heap Inspection Version:1

Categories

OWASP Top 10 2013: A6-Sensitive Data Exposure
FISMA 2014: Media Protection
NIST SP 800-53: SC-4 Information in Shared Resources (P1)
OWASP Top 10 2017: A3-Sensitive Data Exposure

Description

Heap Inspection\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=553
Status	New

Method use_certificate_chain_file at line 437 of openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c defines passwd_callback, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwd_callback, this variable is never cleared from memory.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c
Line	442	442
Object	passwd_callback	passwd_callback

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-3449-FP.c
Method static int use_certificate_chain_file(SSL_CTX *ctx, SSL *ssl, const char *file)

```
....
442.      pem_password_cb *passwd_callback;
```

Heap Inspection\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=554
Status	New

Method use_certificate_chain_file at line 437 of openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c defines passwd_callback, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwd_callback, this variable is never cleared from memory.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c
Line	442	442

Object	passwd_callback	passwd_callback

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-3449-FP.c
Method static int use_certificate_chain_file(SSL_CTX *ctx, SSL *ssl, const char *file)

```
....
442.      pem_password_cb *passwd_callback;
```

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=1020051&projectid=20044&pathid=65
Status	New

The pointer drive_letters at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 1215 is being used after it has been freed.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	1250	1250
Object	drive_letters	drive_letters

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method char RemoveDriveLetters(DWORD DriveIndex, BOOL bReturnLast, BOOL bSilent)

```
....
1250.       return drive_letters[bReturnLast ? (len - 1) : 0];
```

Buffer Overflow Loops

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

Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.2 - Buffer overflows
NIST SP 800-53: SI-16 Memory Protection (P1)
OWASP Top 10 2017: A1-Injection

Description

Buffer Overflow Loops\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=122
Status	New

The buffer allocated by offset in pbatard@@rufus-newest-CVE-2021-3520-FP.c at line 1719 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	1747	1780
Object	1	offset

Code Snippet

File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c

Method const char* GetFsName(HANDLE hPhysical, LARGE_INTEGER StartingOffset)

```
....
1747.       uint8_t* buf = calloc(sector_size, 1);
....
1780.                   if (memcmp(&buf[offset], fat_fs_types[i].magic,
8) == 0)
```

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=1020051&projectid=20044&pathid=135
Status	New

The variable declared in null at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 1565 is not initialized when it is used by type_name at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 1733.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-	openssl@@openssl-openssl-3.2.1-CVE-

	2021-23840-FP.c	2021-23840-FP.c
Line	1697	1735
Object	null	type_name

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static void *evp_cipher_from_algorithm(const int name_id,

```
....
1697.           cipher = NULL;
```

File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method void evp_cipher_free_int(EVP_CIPHER *cipher)

```
....
1735.       OPENSSL_free(cipher->type_name);
```

NULL Pointer Dereference\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=136
Status	New

The variable declared in null at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 1565 is not initialized when it is used by refcnt at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 1741.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	1697	1748
Object	null	refcnt

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static void *evp_cipher_from_algorithm(const int name_id,

```
....
1697.           cipher = NULL;
```

File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method void EVP_CIPHER_free(EVP_CIPHER *cipher)

```
....
1748.       CRYPTO_DOWN_REF(&cipher->refcnt, &i);
```

NULL Pointer Dereference\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=137
Status	New

The variable declared in null at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 1565 is not initialized when it is used by refcnt at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 1733.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Line	1697	1737
Object	null	refcnt

Code Snippet

File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c

Method static void *evp_cipher_from_algorithm(const int name_id,

```
....
1697.           cipher = NULL;
```

File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c

Method void evp_cipher_free_int(EVP_CIPHER *cipher)

```
....
1737.       CRYPTO_FREE_REF(&cipher->refcnt);
```

NULL Pointer Dereference\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=138
Status	New

The variable declared in null at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 1565 is not initialized when it is used by prov at openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c in line 1733.

	Source	Destination
File	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c

Line	1697	1736
Object	null	prov

Code Snippet
File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c
Method static void *evp_cipher_from_algorithm(const int name_id,

```
....
1697.           cipher = NULL;
```

File Name openssl@@openssl-openssl-3.2.1-CVE-2021-23840-FP.c

Method void evp_cipher_free_int(EVP_CIPHER *cipher)

```
....
1736.       ossl_provider_free(cipher->prov);
```

NULL Pointer Dereference\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=139
Status	New

The variable declared in null at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 1565 is not initialized when it is used by type_name at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 1733.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	1697	1735
Object	null	type_name

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method static void *evp_cipher_from_algorithm(const int name_id,

```
....
1697.           cipher = NULL;
```

File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c

Method void evp_cipher_free_int(EVP_CIPHER *cipher)

```
....
1735.       OPENSSL_free(cipher->type_name);
```

NULL Pointer Dereference\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=140
Status	New

The variable declared in null at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 1565 is not initialized when it is used by refcnt at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 1741.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	1697	1748
Object	null	refcnt

Code Snippet

File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c

Method static void *evp_cipher_from_algorithm(const int name_id,

```
....
1697.           cipher = NULL;
```

File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c

Method void EVP_CIPHER_free(EVP_CIPHER *cipher)

```
....
1748.       CRYPTO_DOWN_REF(&cipher->refcnt, &i);
```

NULL Pointer Dereference\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=141
Status	New

The variable declared in null at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 1565 is not initialized when it is used by refcnt at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 1733.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c

Line	1697	1737
Object	null	refcnt

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method static void *evp_cipher_from_algorithm(const int name_id,

```
....
1697.          cipher = NULL;
```

File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c

Method void evp_cipher_free_int(EVP_CIPHER *cipher)

```
....
1737.      CRYPTO_FREE_REF(&cipher->refcnt);
```

NULL Pointer Dereference\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=142
Status	New

The variable declared in null at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 1565 is not initialized when it is used by prov at openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c in line 1733.

	Source	Destination
File	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c	openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Line	1697	1736
Object	null	prov

Code Snippet
File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c
Method static void *evp_cipher_from_algorithm(const int name_id,

```
....
1697.          cipher = NULL;
```

File Name openssl@@openssl-openssl-3.3.1-CVE-2021-23840-FP.c

Method void evp_cipher_free_int(EVP_CIPHER *cipher)

```
....
1736.      ossl_provider_free(cipher->prov);
```

NULL Pointer Dereference\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=143
Status	New

The variable declared in null at OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c in line 2939 is not initialized when it is used by ce at OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c in line 2430.

	Source	Destination
File	OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c
Line	3013	2455
Object	null	ce

Code Snippet

File Name OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c

Method do_init_frame(struct context *c)

```
....
3013.       frame_finalize_options(c, NULL);
```

File Name OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c

Method frame_finalize_options(struct context *c, const struct options *o)

```
....
2455.                       o->ce.tun_mtu);
```

NULL Pointer Dereference\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=144
Status	New

The variable declared in null at OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c in line 2939 is not initialized when it is used by ce at OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c in line 2430.

	Source	Destination
File	OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c
Line	3013	2454
Object	null	ce

Code Snippet
File Name OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c
Method do_init_frame(struct context *c)

```
....
3013.       frame_finalize_options(c, NULL);
```

File Name OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c

Method frame_finalize_options(struct context *c, const struct options *o)

```
....
2454.                       o->ce.tun_mtu_defined,
```

NULL Pointer Dereference\Path 11:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=145
Status New

The variable declared in null at OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c in line 2939 is not initialized when it is used by ce at OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c in line 2430.

	Source	Destination
File	OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c
Line	3013	2453
Object	null	ce

Code Snippet
File Name OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c
Method do_init_frame(struct context *c)

```
....
3013.       frame_finalize_options(c, NULL);
```

File Name OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c

Method frame_finalize_options(struct context *c, const struct options *o)

```
....
2453.                       o->ce.link_mtu,
```

NULL Pointer Dereference\Path 12:

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

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=146
Status	New

The variable declared in null at OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c in line 2939 is not initialized when it is used by ce at OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c in line 2430.

	Source	Destination
File	OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c
Line	3013	2452
Object	null	ce

Code Snippet

File Name OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c

Method do_init_frame(struct context *c)

```
....
3013.       frame_finalize_options(c, NULL);
```

File Name OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c

Method frame_finalize_options(struct context *c, const struct options *o)

```
....
2452.                       o->ce.link_mtu_defined,
```

NULL Pointer Dereference\Path 13:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=147
Status	New

The variable declared in null at OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c in line 3171 is not initialized when it is used by to_link_addr at OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c in line 3171.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Line	3216	3233
Object	null	to_link_addr

Code Snippet

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c

Method tls_multi_process(struct tls_multi *multi,

```
....
3216.               struct link_socket_actual *tla = NULL;
....
3233.                   multi->to_link_addr = *tla;
```

NULL Pointer Dereference\Path 14:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=148
Status	New

The variable declared in null at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788 is not initialized when it is used by prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	791	816
Object	null	prop_array

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL DeletePartition(DWORD DriveIndex, ULONGLONG PartitionOffset, BOOL bSilent)

```
....
791.        VDS_PARTITION_PROP* prop_array = NULL;
....
816.                            prop_array[i].ullOffset,
SizeToHumanReadable(prop_array[i].ullSize, FALSE, FALSE));
```

NULL Pointer Dereference\Path 15:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=149
Status	New

The variable declared in null at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788 is not initialized when it is used by prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	791	816
Object	null	prop_array

Code Snippet	
File Name	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method	BOOL DeletePartition(DWORD DriveIndex, ULONGLONG PartitionOffset, BOOL bSilent)

```
....
791.        VDS_PARTITION_PROP* prop_array = NULL;
....
816.                            prop_array[i].ullOffset,
SizeToHumanReadable(prop_array[i].ullSize, FALSE, FALSE));
```

NULL Pointer Dereference\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=150
Status	New

The variable declared in null at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788 is not initialized when it is used by prop_array at pbatard@@rufus-newest-CVE-2021-3520-FP.c in line 788.

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	791	815
Object	null	prop_array

Code Snippet	
File Name	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method	BOOL DeletePartition(DWORD DriveIndex, ULONGLONG PartitionOffset, BOOL bSilent)

```
....
791.        VDS_PARTITION_PROP* prop_array = NULL;
....
815.                    suprintf("● Partition %d (offset: %lld, size:
%s)", prop_array[i].ulPartitionNumber,
```

NULL Pointer Dereference\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=151
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 855.

	Source	Destination

File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	863
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.            sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method check_timeout_random_component(struct context *c)

```
....
863.            tv_add(&c->c2.timeval, &c->c2.timeout_random_component);
```

NULL Pointer Dereference\Path 18:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=152
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 855.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	863
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.            sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method check_timeout_random_component(struct context *c)

```
....
863.          tv_add(&c->c2.timeval, &c->c2.timeout_random_component);
```

NULL Pointer Dereference\Path 19:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=153
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 855.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	861
Object	0	c2

Code Snippet

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method check_timeout_random_component(struct context *c)

```
....
861.      if (c->c2.timeval.tv_sec >= 1)
```

NULL Pointer Dereference\Path 20:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=154
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 855.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Line	136	857
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method check_timeout_random_component(struct context *c)

```
....
857.      if (now >= c->c2.update_timeout_random_component)
```

NULL Pointer Dereference\Path 21:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=155
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 844.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	851
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method check_timeout_random_component_dowork(struct context *c)

```
....
851.      dmsg(D_INTERVAL, "RANDOM USEC=%ld", (long) c-
>c2.timeout_random_component.tv_usec);
```

NULL Pointer Dereference\Path 22:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=156
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 558.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	579
Object	0	c2

Code Snippet

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method check_fragment(struct context *c)

```
....
579.      fragment_housekeeping(c->c2.fragment, &c->c2.frame_fragment,
&c->c2.timeval);
```

NULL Pointer Dereference\Path 23:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=157
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 558.

	Source	Destination

File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	579
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method check_fragment(struct context *c)

```
....
579.      fragment_housekeeping(c->c2.fragment, &c->c2.frame_fragment,
&c->c2.timeval);
```

NULL Pointer Dereference\Path 24:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=158
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 558.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	579
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method check_fragment(struct context *c)

```
....
579.      fragment_housekeeping(c->c2.fragment, &c->c2.frame_fragment,
&c->c2.timeval);
```

NULL Pointer Dereference\Path 25:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=159
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 558.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	574
Object	0	c2

Code Snippet

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method check_fragment(struct context *c)

```
....
574.              ASSERT(fragment_ready_to_send(c->c2.fragment, &c-
>c2.buf, &c->c2.frame_fragment));
```

NULL Pointer Dereference\Path 26:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=160
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 558.

	Source	Destination

File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	574
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method check_fragment(struct context *c)

```
....
574.              ASSERT(fragment_ready_to_send(c->c2.fragment, &c->c2.buf, &c->c2.frame_fragment));
```

NULL Pointer Dereference\Path 27:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=161
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 558.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	574
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method check_fragment(struct context *c)

```
....
574.              ASSERT(fragment_ready_to_send(c->c2.fragment, &c-
>c2.buf, &c->c2.frame_fragment));
```

NULL Pointer Dereference\Path 28:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=162
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 558.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	571
Object	0	c2

Code Snippet

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method check_fragment(struct context *c)

```
....
571.          if (!c->c2.to_link.len)
```

NULL Pointer Dereference\Path 29:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=163
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 558.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-	OpenVPN@@openvpn-v2.6.5-CVE-2023-

	46849-TP.c	46849-TP.c
Line	136	569
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method check_fragment(struct context *c)

```
....
569.      if (fragment_outgoing_defined(c->c2.fragment))
```

NULL Pointer Dereference\Path 30:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=164
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	686
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method encrypt_sign(struct context *c, bool comp_frag)

```
....
686.      buffer_turnover(orig_buf, &c->c2.to_link, &c->c2.buf, &b-
>read_tun_buf);
```

NULL Pointer Dereference\Path 31:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=165
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	686
Object	0	c2

Code Snippet

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method encrypt_sign(struct context *c, bool comp_frag)

```
....
686.      buffer_turnover(orig_buf, &c->c2.to_link, &c->c2.buf, &b-
>read_tun_buf);
```

NULL Pointer Dereference\Path 32:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=166
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination

File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	683
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method encrypt_sign(struct context *c, bool comp_frag)

```
....
683.                                      &c->c2.to_link_addr);
```

NULL Pointer Dereference\Path 33:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=167
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	682
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method encrypt_sign(struct context *c, bool comp_frag)

```
....
682.      link_socket_get_outgoing_addr(&c->c2.buf,
get_link_socket_info(c),
```

NULL Pointer Dereference\Path 34:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=168
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	676
Object	0	c2

Code Snippet

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method encrypt_sign(struct context *c, bool comp_frag)

```
....
676.          tls_post_encrypt(c->c2.tls_multi, &c->c2.buf);
```

NULL Pointer Dereference\Path 35:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=169
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-	OpenVPN@@openvpn-v2.6.5-CVE-2023-

	46849-TP.c	46849-TP.c
Line	136	676
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.            sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method encrypt_sign(struct context *c, bool comp_frag)

```
....
676.            tls_post_encrypt(c->c2.tls_multi, &c->c2.buf);
```

NULL Pointer Dereference\Path 36:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=170
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	674
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.            sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method encrypt_sign(struct context *c, bool comp_frag)

```
....
674.              tls_prepend_opcode_v1(c->c2.tls_multi, &c->c2.buf);
```

NULL Pointer Dereference\Path 37:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=171
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	674
Object	0	c2

Code Snippet

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method encrypt_sign(struct context *c, bool comp_frag)

```
....
674.              tls_prepend_opcode_v1(c->c2.tls_multi, &c->c2.buf);
```

NULL Pointer Dereference\Path 38:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=172
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Line	136	672
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method encrypt_sign(struct context *c, bool comp_frag)

```
....
672.          if (c->c2.buf.len > 0 && !c->c2.tls_multi->use_peer_id)
```

NULL Pointer Dereference\Path 39:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=173
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	672
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method encrypt_sign(struct context *c, bool comp_frag)

```
....
672.          if (c->c2.buf.len > 0 && !c->c2.tls_multi->use_peer_id)
```

NULL Pointer Dereference\Path 40:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=174
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	670
Object	0	c2

Code Snippet

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method context_reschedule_sec(struct context *c, int sec)

```
....
136.            sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method encrypt_sign(struct context *c, bool comp_frag)

```
....
670.        if (c->c2.tls_multi)
```

NULL Pointer Dereference\Path 41:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=175
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	667
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method encrypt_sign(struct context *c, bool comp_frag)

```
....
667.      openvpn_encrypt(&c->c2.buf, b->encrypt_buf, co);
```

NULL Pointer Dereference\Path 42:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=176
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	658
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method encrypt_sign(struct context *c, bool comp_frag)

```
....
658.              tls_prepend_opcode_v2(c->c2.tls_multi, &b-
>encrypt_buf);
```

NULL Pointer Dereference\Path 43:

Severity	Low
Result State	To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=177
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	656
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.            sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method encrypt_sign(struct context *c, bool comp_frag)

```
....
656.            if (c->c2.buf.len > 0 && c->c2.tls_multi->use_peer_id)
```

NULL Pointer Dereference\Path 44:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=178
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	656
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method encrypt_sign(struct context *c, bool comp_frag)

```
....
656.          if (c->c2.buf.len > 0 && c->c2.tls_multi->use_peer_id)
```

NULL Pointer Dereference\Path 45:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=179
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	652
Object	0	c2

Code Snippet

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method encrypt_sign(struct context *c, bool comp_frag)

```
....
652.          tls_pre_encrypt(c->c2.tls_multi, &c->c2.buf, &co);
```

NULL Pointer Dereference\Path 46:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=180
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	652
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.            sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method encrypt_sign(struct context *c, bool comp_frag)

```
....
652.            tls_pre_encrypt(c->c2.tls_multi, &c->c2.buf, &co);
```

NULL Pointer Dereference\Path 47:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=181
Status New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	649
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.            sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method encrypt_sign(struct context *c, bool comp_frag)

```
....
649.      if (c->c2.tls_multi)
```

NULL Pointer Dereference\Path 48:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=182
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	647
Object	0	c2

Code Snippet

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method encrypt_sign(struct context *c, bool comp_frag)

```
....
647.      ASSERT(buf_init(&b->encrypt_buf, c->c2.frame.buf.headroom));
```

NULL Pointer Dereference\Path 49:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=183
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	641
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method encrypt_sign(struct context *c, bool comp_frag)

```
....
641.              fragment_outgoing(c->c2.fragment, &c->c2.buf, &c-
>c2.frame_fragment);
```

NULL Pointer Dereference\Path 50:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=184
Status	New

The variable declared in 0 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 132 is not initialized when it is used by c2 at OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c in line 606.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	136	641
Object	0	c2

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method context_reschedule_sec(struct context *c, int sec)

```
....
136.          sec = 0;
```

File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c

Method	encrypt_sign(struct context *c, bool comp_frag)

```
....
641.              fragment_outgoing(c->c2.fragment, &c->c2.buf, &c-
>c2.frame_fragment);
```

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=1020051&projectid=20044&pathid=5
Status	New

The *combine_strings method calls the snprintf function, at line 35 of openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.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	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	39	39
Object	snprintf	snprintf

Code Snippet	
File Name	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method	char *combine_strings(const char *string_one, const char *string_two,

```
....
39.    snprintf(combined, combined_len, "%s%c%s", string_one, delimiter,
string_two);
```

Unchecked Return Value\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=6
Status	New

The *addbrackets method calls the strdup function, at line 105 of openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.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	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	118	118
Object	strdup	strdup

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method char *addbrackets(const char *string) {

```
....
118.    return strdup(string);
```

Unchecked Return Value\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=7
Status	New

The read_file method calls the snprintf function, at line 235 of openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.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	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	250	250
Object	snprintf	snprintf

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
250.    snprintf(last_scanned_filename, sizeof(last_scanned_filename),
"%s", file);
```

Unchecked Return Value\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=8
Status	New

The *combine_strings method calls the snprintf function, at line 35 of openSUSE@@libeconf-v0.3.4-CVE-2023-32181-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	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	39	39
Object	snprintf	snprintf

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method char *combine_strings(const char *string_one, const char *string_two,

```
....
39.    snprintf(combined, combined_len, "%s%c%s", string_one, delimiter,
string_two);
```

Unchecked Return Value\Path 5:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=9
Status New

The *addbrackets method calls the strdup function, at line 103 of openSUSE@@libeconf-v0.3.4-CVE-2023-32181-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	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	114	114
Object	strdup	strdup

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method char *addbrackets(const char *string) {

```
....
114.    return strdup(string);
```

Unchecked Return Value\Path 6:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=10

Status	New

The *combine_strings method calls the snprintf function, at line 35 of openSUSE@@libeconf-v0.3.6-CVE-2023-32181-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	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	39	39
Object	snprintf	snprintf

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method char *combine_strings(const char *string_one, const char *string_two,

```
....
39.    snprintf(combined, combined_len, "%s%c%s", string_one, delimiter,
string_two);
```

Unchecked Return Value\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=11
Status	New

The *addbrackets method calls the strdup function, at line 103 of openSUSE@@libeconf-v0.3.6-CVE-2023-32181-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	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	114	114
Object	strdup	strdup

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method char *addbrackets(const char *string) {

```
....
114.    return strdup(string);
```

Unchecked Return Value\Path 8:

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

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=12
Status	New

The *combine_strings method calls the snprintf function, at line 35 of openSUSE@@libeconf-v0.4.5-CVE-2023-32181-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	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	39	39
Object	snprintf	snprintf

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method char *combine_strings(const char *string_one, const char *string_two,

```
....
39.    snprintf(combined, combined_len, "%s%c%s", string_one, delimiter,
string_two);
```

Unchecked Return Value\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=13
Status	New

The *addbrackets method calls the strdup function, at line 105 of openSUSE@@libeconf-v0.4.5-CVE-2023-32181-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	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	118	118
Object	strdup	strdup

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method char *addbrackets(const char *string) {

```
....
118.    return strdup(string);
```

Unchecked Return Value\Path 10:

Severity	Low

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=14
Status	New

The *combine_strings method calls the snprintf function, at line 35 of openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.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	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Line	39	39
Object	snprintf	snprintf

Code Snippet
File Name openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Method char *combine_strings(const char *string_one, const char *string_two,

```
....
39.    snprintf(combined, combined_len, "%s%c%s", string_one, delimiter,
string_two);
```

Unchecked Return Value\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=15
Status	New

The *addbrackets method calls the strdup function, at line 107 of openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.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	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Line	122	122
Object	strdup	strdup

Code Snippet
File Name openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Method char *addbrackets(const char *string) {

```
....
122.    return strdup(string);
```

Unchecked Return Value\Path 12:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=16
Status	New

The read_file method calls the snprintf function, at line 242 of openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.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	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	257	257
Object	snprintf	snprintf

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
257.    snprintf(last_scanned_filename, sizeof(last_scanned_filename),
"%s", file);
```

Unchecked Return Value\Path 13:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=17
Status	New

The jsi_code_decode method calls the snprintf function, at line 434 of pcmacdon@@jsish-3.0-CVE-2020-22873-TP.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	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Line	437	437
Object	snprintf	snprintf

Code Snippet
File Name pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Method void jsi_code_decode(Jsi_Interp *interp, jsi_OpCode *op, int currentip, char *buf, int bsiz)

```
....
437.          snprintf(buf, bsiz, "Bad opcode[%d] at %d", op->op,
currentip);
```

Unchecked Return Value\Path 14:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=18
Status	New

The jsi_code_decode method calls the snprintf function, at line 434 of pcmacdon@@jsish-3.0-CVE-2020-22873-TP.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	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Line	440	440
Object	snprintf	snprintf

Code Snippet
File Name pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Method void jsi_code_decode(Jsi_Interp *interp, jsi_OpCode *op, int currentip, char *buf, int bsiz)

```
....
440.      snprintf(nbuf, sizeof(nbuf), "%d#%d", currentip, op->Line);
```

Unchecked Return Value\Path 15:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=19
Status	New

The jsi_code_decode method calls the snprintf function, at line 434 of pcmacdon@@jsish-3.0-CVE-2020-22873-TP.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	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Line	441	441
Object	snprintf	snprintf

Code Snippet	
File Name	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Method	void jsi_code_decode(Jsi_Interp *interp, jsi_OpCode *op, int currentip, char *buf, int bsiz)

```
....
441.      snprintf(buf, bsiz, "%-8s %s ", nbuf, jsi_op_names[op->op]);
```

Unchecked Return Value\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=20
Status	New

The jsi_code_decode method calls the snprintf function, at line 434 of pcmacdon@@jsish-3.0-CVE-2020-22873-TP.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	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Line	458	458
Object	snprintf	snprintf

Code Snippet	
File Name	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Method	void jsi_code_decode(Jsi_Interp *interp, jsi_OpCode *op, int currentip, char *buf, int bsiz)

```
....
458.               op->op == OP_SCATCH) snprintf(bp, bsiz, "\"%s\"", op->data ? (char*)op->data:"(NoCatch)");
```

Unchecked Return Value\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=21
Status	New

The jsi_code_decode method calls the snprintf function, at line 434 of pcmacdon@@jsish-3.0-CVE-2020-22873-TP.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	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c

Line	459	459
Object	snprintf	snprintf

Code Snippet
File Name pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Method void jsi_code_decode(Jsi_Interp *interp, jsi_OpCode *op, int currentip, char *buf, int bsiz)

```
....
459.      else if (op->op == OP_PUSHVAR) snprintf(bp, bsiz, "var:
\"%s\"", ((jsi_FastVar *)op->data)->varname);
```

Unchecked Return Value\Path 18:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=22
Status	New

The jsi_code_decode method calls the snprintf function, at line 434 of pcmacdon@@jsish-3.0-CVE-2020-22873-TP.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	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Line	466	466
Object	snprintf	snprintf

Code Snippet
File Name pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Method void jsi_code_decode(Jsi_Interp *interp, jsi_OpCode *op, int currentip, char *buf, int bsiz)

```
....
466.          snprintf(bp, bsiz, "{%d},%d\t#%d", jp->off, jp->topop,
currentip + jp->off);
```

Unchecked Return Value\Path 19:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=23
Status	New

The jsi_code_decode method calls the snprintf function, at line 434 of pcmacdon@@jsish-3.0-CVE-2020-22873-TP.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	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c	pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Line	470	470
Object	snprintf	snprintf

Code Snippet
File Name pcmacdon@@jsish-3.0-CVE-2020-22873-TP.c
Method void jsi_code_decode(Jsi_Interp *interp, jsi_OpCode *op, int currentip, char *buf, int bsiz)

```
....
470.          snprintf(bp, bsiz, "{try:%d, catch:%d, final:%d}", t->trylen, t->catchlen, t->finallen);
```

Unchecked Return Value\Path 20:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=24
Status New

The cpy_file_entry method calls the group function, at line 213 of openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.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	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	215	215
Object	group	group

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
215.    copied_fe.group = strdup(fe.group);
```

Unchecked Return Value\Path 21:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=25
Status New

The cpy_file_entry method calls the key function, at line 213 of openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.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	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	216	216
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
216.    copied_fe.key = strdup(fe.key);
```

Unchecked Return Value\Path 22:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=26
Status New

The initialize method calls the group function, at line 44 of openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.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	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	45	45
Object	group	group

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
45.    key_file->file_entry[num].group = strdup(KEY_FILE_NULL_VALUE);
```

Unchecked Return Value\Path 23:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=27
Status New

The initialize method calls the key function, at line 44 of openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.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	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	46	46
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
46.    key_file->file_entry[num].key = strdup(KEY_FILE_NULL_VALUE);
```

Unchecked Return Value\Path 24:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=28
Status New

The initialize method calls the value function, at line 44 of openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.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	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Line	47	47
Object	value	value

Code Snippet
File Name openSUSE@@libeconf-0.4.0-CVE-2023-32181-TP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
47.    key_file->file_entry[num].value = strdup(KEY_FILE_NULL_VALUE);
```

Unchecked Return Value\Path 25:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=29

Status	New

The last_scanned_file method calls the Pointer function, at line 508 of openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.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	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	511	511
Object	Pointer	Pointer

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method void last_scanned_file(char **filename, uint64_t *line_nr)

```
....
511.    *filename = strdup(last_scanned_filename);
```

Unchecked Return Value\Path 26:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=30
Status	New

The join_same_entries method calls the value function, at line 41 of openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.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	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	56	56
Object	value	value

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
56.     ef->file_entry[i].value = strdup("");
```

Unchecked Return Value\Path 27:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20

	044&pathid=31
Status	New

The join_same_entries method calls the comment_after_value function, at line 41 of openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.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	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	104	104
Object	comment_after_value	comment_after_value

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
104.              ef->file_entry[i].comment_after_value = strdup(post);
```

Unchecked Return Value\Path 28:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=32
Status	New

The cpy_file_entry method calls the group function, at line 209 of openSUSE@@libeconf-v0.3.4-CVE-2023-32181-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	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	211	211
Object	group	group

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
211.    copied_fe.group = strdup(fe.group);
```

Unchecked Return Value\Path 29:

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

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=33
Status	New

The cpy_file_entry method calls the key function, at line 209 of openSUSE@@libeconf-v0.3.4-CVE-2023-32181-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	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	212	212
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
212.    copied_fe.key = strdup(fe.key);
```

Unchecked Return Value\Path 30:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=34
Status	New

The initialize method calls the group function, at line 44 of openSUSE@@libeconf-v0.3.4-CVE-2023-32181-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	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	45	45
Object	group	group

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
45.    key_file->file_entry[num].group = strdup(KEY_FILE_NULL_VALUE);
```

Unchecked Return Value\Path 31:

Severity	Low
Result State	To Verify

Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=35
Status	New

The initialize method calls the key function, at line 44 of openSUSE@@libeconf-v0.3.4-CVE-2023-32181-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	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	46	46
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
46.    key_file->file_entry[num].key = strdup(KEY_FILE_NULL_VALUE);
```

Unchecked Return Value\Path 32:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=36
Status	New

The initialize method calls the value function, at line 44 of openSUSE@@libeconf-v0.3.4-CVE-2023-32181-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	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Line	47	47
Object	value	value

Code Snippet
File Name openSUSE@@libeconf-v0.3.4-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
47.    key_file->file_entry[num].value = strdup(KEY_FILE_NULL_VALUE);
```

Unchecked Return Value\Path 33:

Severity	Low

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=37
Status	New

The cpy_file_entry method calls the group function, at line 209 of openSUSE@@libeconf-v0.3.6-CVE-2023-32181-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	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	211	211
Object	group	group

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
211.    copied_fe.group = strdup(fe.group);
```

Unchecked Return Value\Path 34:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=38
Status	New

The cpy_file_entry method calls the key function, at line 209 of openSUSE@@libeconf-v0.3.6-CVE-2023-32181-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	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	212	212
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
212.    copied_fe.key = strdup(fe.key);
```

Unchecked Return Value\Path 35:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=39
Status	New

The initialize method calls the group function, at line 44 of openSUSE@@libeconf-v0.3.6-CVE-2023-32181-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	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	45	45
Object	group	group

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
45.     key_file->file_entry[num].group = strdup(KEY_FILE_NULL_VALUE);
```

Unchecked Return Value\Path 36:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=40
Status	New

The initialize method calls the key function, at line 44 of openSUSE@@libeconf-v0.3.6-CVE-2023-32181-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	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	46	46
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
46.     key_file->file_entry[num].key = strdup(KEY_FILE_NULL_VALUE);
```

Unchecked Return Value\Path 37:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=41
Status	New

The initialize method calls the value function, at line 44 of openSUSE@@libeconf-v0.3.6-CVE-2023-32181-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	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Line	47	47
Object	value	value

Code Snippet
File Name openSUSE@@libeconf-v0.3.6-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
47.     key_file->file_entry[num].value = strdup(KEY_FILE_NULL_VALUE);
```

Unchecked Return Value\Path 38:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=42
Status	New

The cpy_file_entry method calls the group function, at line 213 of openSUSE@@libeconf-v0.4.5-CVE-2023-32181-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	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	215	215
Object	group	group

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
215.     copied_fe.group = strdup(fe.group);
```

Unchecked Return Value\Path 39:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=43
Status	New

The cpy_file_entry method calls the key function, at line 213 of openSUSE@@libeconf-v0.4.5-CVE-2023-32181-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	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	216	216
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
216.    copied_fe.key = strdup(fe.key);
```

Unchecked Return Value\Path 40:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=44
Status	New

The initialize method calls the group function, at line 44 of openSUSE@@libeconf-v0.4.5-CVE-2023-32181-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	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	45	45
Object	group	group

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
45.    key_file->file_entry[num].group = strdup(KEY_FILE_NULL_VALUE);
```

Unchecked Return Value\Path 41:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=45
Status	New

The initialize method calls the key function, at line 44 of openSUSE@@libeconf-v0.4.5-CVE-2023-32181-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	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	46	46
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
46.    key_file->file_entry[num].key = strdup(KEY_FILE_NULL_VALUE);
```

Unchecked Return Value\Path 42:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=46
Status	New

The initialize method calls the value function, at line 44 of openSUSE@@libeconf-v0.4.5-CVE-2023-32181-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	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c	openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c
Line	47	47
Object	value	value

Code Snippet
File Name openSUSE@@libeconf-v0.4.5-CVE-2023-32181-FP.c

Method	void initialize(econf_file *key_file, size_t num) {

```
....
47.    key_file->file_entry[num].value = strdup(KEY_FILE_NULL_VALUE);
```

Unchecked Return Value\Path 43:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=47
Status	New

The cpy_file_entry method calls the group function, at line 215 of openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.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	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Line	217	217
Object	group	group

Code Snippet	
File Name	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Method	struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
217.    copied_fe.group = strdup(fe.group);
```

Unchecked Return Value\Path 44:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=48
Status	New

The cpy_file_entry method calls the key function, at line 215 of openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.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	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Line	218	218
Object	key	key

Code Snippet

File Name	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Method	struct file_entry cpy_file_entry(struct file_entry fe) {

```
....
218.    copied_fe.key = strdup(fe.key);
```

Unchecked Return Value\Path 45:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=49
Status	New

The initialize method calls the group function, at line 44 of openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.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	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Line	45	45
Object	group	group

Code Snippet	
File Name	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Method	void initialize(econf_file *key_file, size_t num) {

```
....
45.    key_file->file_entry[num].group = strdup(KEY_FILE_NULL_VALUE);
```

Unchecked Return Value\Path 46:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=50
Status	New

The initialize method calls the key function, at line 44 of openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.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	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Line	46	46
Object	key	key

Code Snippet
File Name openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
46.    key_file->file_entry[num].key = strdup(KEY_FILE_NULL_VALUE);
```

Unchecked Return Value\Path 47:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=51
Status	New

The initialize method calls the value function, at line 44 of openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.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	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Line	47	47
Object	value	value

Code Snippet
File Name openSUSE@@libeconf-v0.4.7-CVE-2023-32181-TP.c
Method void initialize(econf_file *key_file, size_t num) {

```
....
47.    key_file->file_entry[num].value = strdup(KEY_FILE_NULL_VALUE);
```

Unchecked Return Value\Path 48:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=52
Status	New

The last_scanned_file method calls the Pointer function, at line 536 of openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.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	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	539	539
Object	Pointer	Pointer

Code Snippet	
File Name	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method	void last_scanned_file(char **filename, uint64_t *line_nr)

```
....
539.    *filename = strdup(last_scanned_filename);
```

Unchecked Return Value\Path 49:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=53
Status	New

The join_same_entries method calls the value function, at line 41 of openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.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	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	56	56
Object	value	value

Code Snippet	
File Name	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method	join_same_entries(econf_file *ef)

```
....
56.     ef->file_entry[i].value = strdup("");
```

Unchecked Return Value\Path 50:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=54
Status	New

The join_same_entries method calls the comment_after_value function, at line 41 of openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.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	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	104	104

Object	comment_after_value	comment_after_value

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method join_same_entries(econf_file *ef)

```
....
104.              ef->file_entry[i].comment_after_value = strdup(post);
```

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=1020051&projectid=20044&pathid=746
Status New

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	261	261
Object	fgets	fgets

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
261.    while (fgets(buf, sizeof(buf), kf)) {
```

Improper Resource Access Authorization\Path 2:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=747
Status New

	Source	Destination

File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	268	268
Object	fgets	fgets

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
268.    while (fgets(buf, sizeof(buf), kf)) {
```

Improper Resource Access Authorization\Path 3:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=748
Status New

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	261	261
Object	buf	buf

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
261.    while (fgets(buf, sizeof(buf), kf)) {
```

Improper Resource Access Authorization\Path 4:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=749
Status New

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	268	268
Object	buf	buf

Code Snippet	
File Name	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method	read_file(econf_file *ef, const char *file,

```
....
268.    while (fgets(buf, sizeof(buf), kf)) {
```

Improper Resource Access Authorization\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=750
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Line	910	910
Object	fprintf	fprintf

Code Snippet	
File Name	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Method	process_incoming_link_part1(struct context *c, struct link_socket_info *lsi, bool floated)

```
....
910.            fprintf(stderr, "R");
```

Improper Resource Access Authorization\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=751
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Line	1276	1276
Object	fprintf	fprintf

Code Snippet	
File Name	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Method	process_incoming_tun(struct context *c)

```
....
1276.          fprintf(stderr, "r");
```

Improper Resource Access Authorization\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=752
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Line	1604	1604
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Method process_outgoing_link(struct context *c)

```
....
1604.                  fprintf(stderr, "W");
```

Improper Resource Access Authorization\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=753
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Line	1739	1739
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.0-CVE-2023-46849-TP.c
Method process_outgoing_tun(struct context *c)

```
....
1739.              fprintf(stderr, "w");
```

Improper Resource Access Authorization\Path 9:

Severity	Low

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=754
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c
Line	910	910
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c
Method process_incoming_link_part1(struct context *c, struct link_socket_info *lsi, bool floated)

```
....
910.          fprintf(stderr, "R");
```

Improper Resource Access Authorization\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=755
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c
Line	1276	1276
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c
Method process_incoming_tun(struct context *c)

```
....
1276.          fprintf(stderr, "r");
```

Improper Resource Access Authorization\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=756
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c
Line	1604	1604
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c
Method process_outgoing_link(struct context *c)

```
....
1604.                   fprintf(stderr, "W");
```

Improper Resource Access Authorization\Path 12:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=757
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c
Line	1739	1739
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.1-CVE-2023-46849-TP.c
Method process_outgoing_tun(struct context *c)

```
....
1739.               fprintf(stderr, "w");
```

Improper Resource Access Authorization\Path 13:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=758
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Line	911	911

Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Method process_incoming_link_part1(struct context *c, struct link_socket_info *lsi, bool floated)

```
....
911.          fprintf(stderr, "R");
```

Improper Resource Access Authorization\Path 14:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=759
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Line	1278	1278
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Method process_incoming_tun(struct context *c)

```
....
1278.          fprintf(stderr, "r");
```

Improper Resource Access Authorization\Path 15:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=760
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Line	1606	1606
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Method process_outgoing_link(struct context *c)

```
....
1606.                   fprintf(stderr, "W");
```

Improper Resource Access Authorization\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=761
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Line	1741	1741
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.3-CVE-2023-46849-TP.c
Method process_outgoing_tun(struct context *c)

```
....
1741.               fprintf(stderr, "w");
```

Improper Resource Access Authorization\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=762
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Line	911	911
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Method process_incoming_link_part1(struct context *c, struct link_socket_info *lsi, bool floated)

```
....
911.           fprintf(stderr, "R");
```

Improper Resource Access Authorization\Path 18:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=763
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Line	1278	1278
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Method process_incoming_tun(struct context *c)

```
....
1278.           fprintf(stderr, "r");
```

Improper Resource Access Authorization\Path 19:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=764
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Line	1606	1606
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Method process_outgoing_link(struct context *c)

```
....
1606.                   fprintf(stderr, "W");
```

Improper Resource Access Authorization\Path 20:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=765
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Line	1741	1741
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.4-CVE-2023-46849-TP.c
Method process_outgoing_tun(struct context *c)

```
....
1741.               fprintf(stderr, "w");
```

Improper Resource Access Authorization\Path 21:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=766
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Line	911	911
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Method process_incoming_link_part1(struct context *c, struct link_socket_info *lsi, bool floated)

```
....
911.           fprintf(stderr, "R");
```

Improper Resource Access Authorization\Path 22:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=767
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Line	1278	1278

Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Method process_incoming_tun(struct context *c)

```
....
1278.           fprintf(stderr, "r");
```

Improper Resource Access Authorization\Path 23:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=768
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Line	1606	1606
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Method process_outgoing_link(struct context *c)

```
....
1606.                   fprintf(stderr, "W");
```

Improper Resource Access Authorization\Path 24:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=769
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Line	1741	1741
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.6-CVE-2023-46849-TP.c
Method process_outgoing_tun(struct context *c)

```
....
1741.               fprintf(stderr, "w");
```

Improper Resource Access Authorization\Path 25:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=770
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Line	911	911
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Method process_incoming_link_part1(struct context *c, struct link_socket_info *lsi, bool floated)

```
....
911.           fprintf(stderr, "R");
```

Improper Resource Access Authorization\Path 26:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=771
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Line	1278	1278
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Method process_incoming_tun(struct context *c)

```
....
1278.           fprintf(stderr, "r");
```

Improper Resource Access Authorization\Path 27:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=772
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Line	1606	1606
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Method process_outgoing_link(struct context *c)

```
....
1606.                   fprintf(stderr, "W");
```

Improper Resource Access Authorization\Path 28:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=773
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Line	1748	1748
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.8-CVE-2023-46849-FP.c
Method process_outgoing_tun(struct context *c)

```
....
1748.               fprintf(stderr, "w");
```

Improper Resource Access Authorization\Path 29:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=774
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Line	911	911
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Method process_incoming_link_part1(struct context *c, struct link_socket_info *lsi, bool floated)

```
....
911.          fprintf(stderr, "R");
```

Improper Resource Access Authorization\Path 30:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=775
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Line	1278	1278
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Method process_incoming_tun(struct context *c)

```
....
1278.          fprintf(stderr, "r");
```

Improper Resource Access Authorization\Path 31:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=776
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Line	1606	1606

Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Method process_outgoing_link(struct context *c)

```
....
1606.                   fprintf(stderr, "W");
```

Improper Resource Access Authorization\Path 32:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=777
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Line	1746	1746
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.5.9-CVE-2023-46849-FP.c
Method process_outgoing_tun(struct context *c)

```
....
1746.               fprintf(stderr, "w");
```

Improper Resource Access Authorization\Path 33:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=778
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Line	4984	4984
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Method write_pid_file(const char *filename, const char *chroot_dir)

```
....
4984.           fprintf(fp, "%u\n", pid);
```

Improper Resource Access Authorization\Path 34:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=779
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	1025	1025
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method process_incoming_link_part1(struct context *c, struct link_socket_info *lsi, bool floated)

```
....
1025.           fprintf(stderr, "R");
```

Improper Resource Access Authorization\Path 35:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=780
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	1439	1439
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method process_incoming_tun(struct context *c)

```
....
1439.           fprintf(stderr, "r");
```

Improper Resource Access Authorization\Path 36:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=781
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	1767	1767
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method process_outgoing_link(struct context *c)

```
....
1767.                   fprintf(stderr, "W");
```

Improper Resource Access Authorization\Path 37:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=782
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Line	1903	1903
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46849-TP.c
Method process_outgoing_tun(struct context *c)

```
....
1903.               fprintf(stderr, "w");
```

Improper Resource Access Authorization\Path 38:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=783
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c
Line	5000	5000
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c
Method write_pid_file(const char *filename, const char *chroot_dir)

```
....
5000.           fprintf(fp, "%u\n", pid);
```

Improper Resource Access Authorization\Path 39:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=784
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c
Line	4989	4989
Object	fprintf	fprintf

Code Snippet
File Name OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c
Method write_pid_file(const char *filename, const char *chroot_dir)

```
....
4989.           fprintf(fp, "%u\n", pid);
```

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=1020051&projectid=20044&pathid=357
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Line	1241	1241
Object	dest	dest

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Method move_session(struct tls_multi *multi, int dest, int src, bool reinit_src)

```
....
1241.       multi->session[dest] = multi->session[src];
```

Unchecked Array Index\Path 2:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=358
Status New

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	1195	1195
Object	i	i

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method static BOOL _GetDriveLettersAndType(DWORD DriveIndex, char* drive_letters, UINT* drive_type)

```
....
1195.               drive_letters[i] = 0;
```

Unchecked Array Index\Path 3:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=359
Status New

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c

Line	2316	2316
Object	PI_ESP	PI_ESP

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL CreatePartition(HANDLE hDrive, int partition_style, int file_system, BOOL mbr_uefi_marker, uint8_t extra_partitions)

```
....
2316.              partition_index[PI_ESP] = pi;
```

Unchecked Array Index\Path 4:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=360
Status New

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	2355	2355
Object	PI_CASPER	PI_CASPER

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL CreatePartition(HANDLE hDrive, int partition_style, int file_system, BOOL mbr_uefi_marker, uint8_t extra_partitions)

```
....
2355.                    partition_index[PI_CASPER] = pi;
```

Unchecked Array Index\Path 5:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=361
Status New

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	2360	2360
Object	PI_ESP	PI_ESP

Code Snippet	
File Name	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method	BOOL CreatePartition(HANDLE hDrive, int partition_style, int file_system, BOOL mbr_uefi_marker, uint8_t extra_partitions)

```
....
2360.                   partition_index[PI_ESP] = pi;
```

Unchecked Array Index\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=362
Status	New

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	2364	2364
Object	PI_UEFI_NTFS	PI_UEFI_NTFS

Code Snippet	
File Name	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method	BOOL CreatePartition(HANDLE hDrive, int partition_style, int file_system, BOOL mbr_uefi_marker, uint8_t extra_partitions)

```
....
2364.                   partition_index[PI_UEFI_NTFS] = pi;
```

Unchecked Array Index\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=363
Status	New

	Source	Destination
File	osquery@@osquery-4.3.0-CVE-2020-26273-TP.c	osquery@@osquery-4.3.0-CVE-2020-26273-TP.c
Line	48	48
Object	column_name	column_name

Code Snippet	
File Name	osquery@@osquery-4.3.0-CVE-2020-26273-TP.c
Method	Status genSqliteTableRow(sqlite3_stmt* stmt,

```
....
48.          r[column_name] = std::string(reinterpret_cast<const
char*>(text_value));
```

Unchecked Array Index\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=364
Status	New

	Source	Destination
File	osquery@@osquery-4.3.0-CVE-2020-26273-TP.c	osquery@@osquery-4.3.0-CVE-2020-26273-TP.c
Line	54	54
Object	column_name	column_name

Code Snippet
File Name osquery@@osquery-4.3.0-CVE-2020-26273-TP.c
Method Status genSqliteTableRow(sqlite3_stmt* stmt,

```
....
54.        r[column_name] = DOUBLE(float_value);
```

Unchecked Array Index\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=365
Status	New

	Source	Destination
File	osquery@@osquery-4.3.0-CVE-2020-26273-TP.c	osquery@@osquery-4.3.0-CVE-2020-26273-TP.c
Line	59	59
Object	column_name	column_name

Code Snippet
File Name osquery@@osquery-4.3.0-CVE-2020-26273-TP.c
Method Status genSqliteTableRow(sqlite3_stmt* stmt,

```
....
59.        r[column_name] = INTEGER(int_value);
```

Unchecked Array Index\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=366
Status	New

	Source	Destination
File	osquery@@osquery-4.5.0-CVE-2020-26273-TP.c	osquery@@osquery-4.5.0-CVE-2020-26273-TP.c
Line	50	50
Object	column_name	column_name

Code Snippet
File Name osquery@@osquery-4.5.0-CVE-2020-26273-TP.c
Method Status genSqliteTableRow(sqlite3_stmt* stmt,

```
....
50.          r[column_name] = std::string(reinterpret_cast<const
char*>(text_value));
```

Unchecked Array Index\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=367
Status	New

	Source	Destination
File	osquery@@osquery-4.5.0-CVE-2020-26273-TP.c	osquery@@osquery-4.5.0-CVE-2020-26273-TP.c
Line	56	56
Object	column_name	column_name

Code Snippet
File Name osquery@@osquery-4.5.0-CVE-2020-26273-TP.c
Method Status genSqliteTableRow(sqlite3_stmt* stmt,

```
....
56.          r[column_name] = DOUBLE(float_value);
```

Unchecked Array Index\Path 12:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=368
Status	New

	Source	Destination
File	osquery@@osquery-4.5.0-CVE-2020-26273-TP.c	osquery@@osquery-4.5.0-CVE-2020-26273-TP.c
Line	61	61
Object	column_name	column_name

Code Snippet
File Name osquery@@osquery-4.5.0-CVE-2020-26273-TP.c
Method Status genSqliteTableRow(sqlite3_stmt* stmt,

```
....
61.         r[column_name] = INTEGER(int_value);
```

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=1020051&projectid=20044&pathid=785
Status	New

	Source	Destination
File	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	245	245
Object	kf	kf

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
245.     FILE *kf = fopen(file, "rbe");
```

Incorrect Permission Assignment For Critical Resources\Path 2:

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

	PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=786
Status	New

	Source	Destination
File	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	252	252
Object	kf	kf

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
252.    FILE *kf = fopen(file, "rbe");
```

Incorrect Permission Assignment For Critical Resources\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=787
Status	New

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	95	95
Object	CreateFileA	CreateFileA

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL SetAutoMount(BOOL enable)

```
....
95.   hMountMgr = CreateFileA(MOUNTMGR_DOS_DEVICE_NAME, 0,
FILE_SHARE_READ|FILE_SHARE_WRITE, NULL, OPEN_EXISTING,
FILE_ATTRIBUTE_NORMAL, NULL);
```

Incorrect Permission Assignment For Critical Resources\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=788
Status	New

	Source	Destination

File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	111	111
Object	CreateFileA	CreateFileA

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL GetAutoMount(BOOL* enabled)

```
....
111.        hMountMgr = CreateFileA(MOUNTMGR_DOS_DEVICE_NAME, 0,
FILE_SHARE_READ|FILE_SHARE_WRITE, NULL, OPEN_EXISTING,
FILE_ATTRIBUTE_NORMAL, NULL);
```

Incorrect Permission Assignment For Critical Resources\Path 5:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=789
Status New

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	158	158
Object	CreateFileA	CreateFileA

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method static HANDLE GetHandle(char* Path, BOOL bLockDrive, BOOL bWriteAccess, BOOL bWriteShare)

```
....
158.              hDrive = CreateFileA(Path,
GENERIC_READ|(bWriteAccess?GENERIC_WRITE:0),
```

Incorrect Permission Assignment For Critical Resources\Path 6:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=790
Status New

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c

Line	308	308
Object	CreateFileWithTimeout	CreateFileWithTimeout

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method char* GetLogicalName(DWORD DriveIndex, uint64_t PartitionOffset, BOOL bKeepTrailingBackslash, BOOL bSilent)

```
....
308.              hDrive = CreateFileWithTimeout(volume_name,
GENERIC_READ, FILE_SHARE_READ|FILE_SHARE_WRITE,
```

Incorrect Permission Assignment For Critical Resources\Path 7:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=791
Status New

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	1150	1150
Object	CreateFileWithTimeout	CreateFileWithTimeout

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method static BOOL _GetDriveLettersAndType(DWORD DriveIndex, char* drive_letters, UINT* drive_type)

```
....
1150.             hDrive = CreateFileWithTimeout(logical_drive,
GENERIC_READ, FILE_SHARE_READ | FILE_SHARE_WRITE,
```

Incorrect Permission Assignment For Critical Resources\Path 8:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=792
Status New

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	2039	2039
Object	CreateFileA	CreateFileA

Code Snippet	
File Name	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method	static BOOL FlushDrive(char drive_letter)

```
....
2039.       hDrive = CreateFileA(logical_drive,
GENERIC_READ|GENERIC_WRITE, FILE_SHARE_READ|FILE_SHARE_WRITE,
```

Use of Insufficiently Random Values

Query Path:
CPP\Cx\CPP Low Visibility\Use of Insufficiently Random Values Version:0

Categories

FISMA 2014: Media Protection
NIST SP 800-53: SC-28 Protection of Information at Rest (P1)
OWASP Top 10 2017: A3-Sensitive Data Exposure

Description

Use of Insufficiently Random Values\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=1
Status	New

Method init_static at line 730 of OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c uses a weak method srandom to produce random values. These values might be used for secret values, personal identifiers or cryptographic input, allowing an attacker to guess the value.

	Source	Destination
File	OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c
Line	749	749
Object	srandom	srandom

Code Snippet	
File Name	OpenVPN@@openvpn-v2.4.9-CVE-2023-46849-FP.c
Method	init_static(void)

```
....
749.            srandom(seed);
```

Use of Insufficiently Random Values\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=2
Status	New

Method init_static at line 814 of OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c uses a weak method srandom to produce random values. These values might be used for secret values, personal identifiers or cryptographic input, allowing an attacker to guess the value.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Line	833	833
Object	srandom	srandom

Code Snippet
File Name OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Method init_static(void)

```
....
833.            srandom(seed);
```

Use of Insufficiently Random Values\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=3
Status	New

Method init_static at line 820 of OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c uses a weak method srandom to produce random values. These values might be used for secret values, personal identifiers or cryptographic input, allowing an attacker to guess the value.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c
Line	839	839
Object	srandom	srandom

Code Snippet
File Name OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c
Method init_static(void)

```
....
839.            srandom(seed);
```

Use of Insufficiently Random Values\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=4
Status	New

Method init_static at line 820 of OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c uses a weak method srandom to produce random values. These values might be used for secret values, personal identifiers or cryptographic input, allowing an attacker to guess the value.

	Source	Destination
File	OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c
Line	839	839
Object	srandom	srandom

Code Snippet
File Name OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c
Method init_static(void)

```
....
839.          srandom(seed);
```

Information Exposure Through Comments

Query Path:
CPP\Cx\CPP Low Visibility\Information Exposure Through Comments Version:1

Categories

FISMA 2014: Identification And Authentication
NIST SP 800-53: SC-28 Protection of Information at Rest (P1)

Description

Information Exposure Through Comments\Path 1:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=795
Status New

	Source	Destination
File	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Line	3487	3487
Object	cipher "n	cipher "n

Code Snippet
File Name OpenVPN@@openvpn-v2.6.11-CVE-2023-46849-FP.c
Method /* Initialise key_type with auth/cipher "none", so the key_type struct is

```
....
3487.      /* Initialise key_type with auth/cipher "none", so the
key_type struct is
```

Information Exposure Through Comments\Path 2:

Severity Low

Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=796
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c	OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Line	281	281
Object	cipher (m	cipher (m

Code Snippet
File Name OpenVPN@@openvpn-v2.6.5-CVE-2023-46850-TP.c
Method * @param cipher The current cipher (may be NULL).

```
....
281.    * @param cipher         The current cipher (may be NULL).
```

Information Exposure Through Comments\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=797
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c
Line	3504	3504
Object	cipher "n	cipher "n

Code Snippet
File Name OpenVPN@@openvpn-v2.6.7-CVE-2023-46849-FP.c
Method /* Initialise key_type with auth/cipher "none", so the key_type struct is

```
....
3504.      /* Initialise key_type with auth/cipher "none", so the
key_type struct is
```

Information Exposure Through Comments\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=798
Status	New

	Source	Destination
File	OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c	OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c
Line	3493	3493
Object	cipher "n	cipher "n

Code Snippet
File Name OpenVPN@@openvpn-v2.6.9-CVE-2023-46849-FP.c
Method /* Initialise key_type with auth/cipher "none", so the key_type struct is

```
....
3493.       /* Initialise key_type with auth/cipher "none", so the
key_type struct is
```

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=1020051&projectid=20044&pathid=355
Status	New

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	2108	2108
Object	mounted_letter	sizeof

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL MountVolume(char* drive_name, char *volume_name)

```
....
2108.       if ( (GetVolumePathNamesForVolumeNameA(volume_name,
mounted_letter, sizeof(mounted_letter), &size))
```

Sizeof Pointer Argument\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20044&pathid=356
Status	New

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	1330	1330
Object	VolumeLabel	sizeof

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL GetDriveLabel(DWORD DriveIndex, char* letters, char** label, BOOL bSilent)

```
....
1330.                   wchar_to_utf8_no_alloc(VolumeName, VolumeLabel,
sizeof(VolumeLabel));
```

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=1020051&projectid=20044&pathid=793
Status New

The read_file method in openSUSE@@libeconf-0.4.2-CVE-2023-32181-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	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c	openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Line	245	245
Object	fopen	fopen

Code Snippet
File Name openSUSE@@libeconf-0.4.2-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
245.    FILE *kf = fopen(file, "rbe");
```

TOCTOU\Path 2:

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020051&projectid=20

	044&pathid=794
Status	New

The read_file method in openSUSE@@libeconf-v0.5.1-CVE-2023-32181-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	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c	openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Line	252	252
Object	fopen	fopen

Code Snippet
File Name openSUSE@@libeconf-v0.5.1-CVE-2023-32181-TP.c
Method read_file(econf_file *ef, const char *file,

```
....
252.    FILE *kf = fopen(file, "rbe");
```

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=1020051&projectid=20044&pathid=354
Status	New

	Source	Destination
File	pbatard@@rufus-newest-CVE-2021-3520-FP.c	pbatard@@rufus-newest-CVE-2021-3520-FP.c
Line	2545	2545
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name pbatard@@rufus-newest-CVE-2021-3520-FP.c
Method BOOL CreatePartition(HANDLE hDrive, int partition_style, int file_system, BOOL mbr_uefi_marker, uint8_t extra_partitions)

```
....
2545.       size = sizeof(DriveLayoutEx) - ((partition_style ==
PARTITION_STYLE_GPT) ?
```

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

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
//..
```

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

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));
    }
}
```

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;
}
```

Buffer Overflow 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
```

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);
```

Char Overflow

Risk

What might happen

Assigning large data types into smaller data types, without proper checks and explicit casting, will lead to undefined behavior and unintentional effects, such as data corruption (e.g. value wraparound, wherein maximum values become minimum values); system crashes; infinite loops; logic errors, such as bypassing of security mechanisms; or even buffer overflows leading to arbitrary code execution.

Cause

How does it happen

This flaw can occur when implicitly casting numerical data types of a larger size, into a variable with a data type of a smaller size. This forces the program to discard some bits of information from the number. Depending on how the numerical data types are stored in memory, this is often the bits with the highest value, causing substantial corruption of the stored number. Alternatively, the sign bit of a signed integer could be lost, completely reversing the intention of the number.

General Recommendations

How to avoid it

- Avoid casting larger data types to smaller types.
- Prefer promoting the target variable to a large enough data type.
- If downcasting is necessary, always check that values are valid and in range of the target type, before casting

---

Source Code Examples

CPP

Unsafe Downsize Casting

```
int unsafe_addition(short op1, int op2) {

    // op2 gets forced from int into a short
    short total = op1 + op2;

    return total;
}
```

Safer Use of Proper Data Types

```
int safe_addition(short op1, int op2) {

    // total variable is of type int, the largest type that is needed
    int total = 0;

    // check if total will overflow available integer size
    if (INT_MAX - abs(op2) > op1)
```

```
    {
        total = op1 + op2;
    }
    else
    {
        // instead of overflow, saturate (but this is not always a good thing)
        total = INT_MAX
    }

    return total;
}
```

Integer Overflow

Risk

What might happen

Assigning large data types into smaller data types, without proper checks and explicit casting, will lead to undefined behavior and unintentional effects, such as data corruption (e.g. value wraparound, wherein maximum values become minimum values); system crashes; infinite loops; logic errors, such as bypassing of security mechanisms; or even buffer overflows leading to arbitrary code execution.

Cause

How does it happen

This flaw can occur when implicitly casting numerical data types of a larger size, into a variable with a data type of a smaller size. This forces the program to discard some bits of information from the number. Depending on how the numerical data types are stored in memory, this is often the bits with the highest value, causing substantial corruption of the stored number. Alternatively, the sign bit of a signed integer could be lost, completely reversing the intention of the number.

General Recommendations

How to avoid it

- Avoid casting larger data types to smaller types.
- Prefer promoting the target variable to a large enough data type.
- If downcasting is necessary, always check that values are valid and in range of the target type, before casting

---

Source Code Examples

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;
}
```

Heap Inspection

Risk

What might happen

All variables stored by the application in unencrypted memory can potentially be retrieved by an unauthorized user, with privlieged access to the machine. For example, a privileged attacker could attach a debugger to the running process, or retrieve the process's memory from the swapfile or crash dump file.
Once the attacker finds the user passwords in memory, these can be reused to easily impersonate the user to the system.

Cause

How does it happen

String variables are immutable - in other words, once a string variable is assigned, its value cannot be changed or removed. Thus, these strings may remain around in memory, possibly in multiple locations, for an indefinite period of time until the garbage collector happens to remove it. Sensitive data, such as passwords, will remain exposed in memory as plaintext with no control over their lifetime.

General Recommendations

How to avoid it

Generic Guidance:

- Do not store senstiive data, such as passwords or encryption keys, in memory in plaintext, even for a short period of time.
- Prefer to use specialized classes that store encrypted memory.
- Alternatively, store secrets temporarily in mutable data types, such as byte arrays, and then promptly zeroize the memory locations.

Specific Recommendations - Java:

- Instead of storing passwords in immutable strings, prefer to use an encrypted memory object, such as SealedObject.

Specific Recommendations - .NET:

- Instead of storing passwords in immutable strings, prefer to use an encrypted memory object, such as SecureString or ProtectedData.

Source Code Examples

Java

Plaintext Password in Immutable String

```
class Heap_Inspection
{
  private string password;
```

```
  void setPassword()
 {
      password = System.console().readLine("Enter your password: ");
  }
}
```

Password Protected in Memory

```
class Heap_Inspection_Fixed
{
  private SealedObject password;

  void setPassword()
 {
      byte[] sKey = getKeyFromConfig();
      Cipher c = Cipher.getInstance("AES");
      c.init(Cipher.ENCRYPT_MODE, sKey);

      char[] input = System.console().readPassword("Enter your password: ");
      password = new SealedObject(Arrays.asList(input), c);

    //Zero out the possible password, for security.
    Arrays.fill(password, '0');
  }
}
```

CPP

Vulnerable C code

```
/* Vulnerable to heap inspection */

#include <stdio.h>


void somefunc(){
      printf("Yea, I'm just being called for the heap of it..\n");
}

void authfunc(){
        char* password = (char *) malloc(256);
        char ch;
        ssize_t k;
            int i=0;
        while(k = read(0, &ch, 1) > 0)
        {
                if (ch == '\n'){
                        password[i]='\0';
                        break;
                } else{
                        password[i++]=ch;
                        fflush(0);
                }
        }
        printf("Password: %s\n",&password[0]);
}
```

```
int main()
{
    printf("Please enter a password:\n");

      authfunc();
      printf("You can now dump memory to find this password!");
      somefunc();
      gets();

}
```

Safe C code

```
/* Pesumably safe heap */

#include <stdio.h>
#include <string.h>

#define STDIN_FILENO 0

void somefunc(){
        printf("Yea, I'm just being called for the heap of it..\n");
}

void authfunc(){
      char* password = (char*) malloc(256);
      int i=0;
      char ch;
      ssize_t k;
      while(k = read(STDIN_FILENO, &ch, 1) > 0)
      {
            if (ch == '\n'){
                  password[i]='\0';
                  break;
            } else{
                  password[i++]=ch;
                  fflush(0);
            }
      }
      i=0;
      memset(password,'\0',256);
}

int main()
{
      printf("Please enter a password:\n");
      authfunc();
      somefunc();
      char ch;
      while(read(STDIN_FILENO, &ch, 1) > 0)
      {
            if (ch == '\n')
                  break;
      }
}
```

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

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

Use of Insufficiently Random Values

Risk

What might happen

Random values are often used as a mechanism to prevent malicious users from guessing a value, such as a password, encryption key, or session identifier. Depending on what this random value is used for, an attacker would be able to predict the next numbers generated, or previously generated values. This could enable the attacker to hijack another user's session, impersonate another user, or crack an encryption key (depending on what the pseudo-random value was used for).

Cause

How does it happen

The application uses a weak method of generating pseudo-random values, such that other numbers could be determined from a relatively small sample size. Since the pseudo-random number generator used is designed for statistically uniform distribution of values, it is approximately deterministic. Thus, after collecting a few generated values (e.g. by creating a few individual sessions, and collecting the sessionids), it would be possible for an attacker to calculate another sessionid.
Specifically, if this pseudo-random value is used in any security context, such as passwords, keys, or secret identifiers, an attacker would be able to predict the next numbers generated, or previously generated values.

General Recommendations

How to avoid it

Generic Guidance:

- Whenever unpredicatable numbers are required in a security context, use a cryptographically strong random number generator, instead of a statistical pseudo-random generator.
- Use the cryptorandom generator that is built-in to your language or platform, and ensure it is securely seeded. Do not seed the generator with a weak, non-random seed. (In most cases, the default is securely random).
- Ensure you use a long enough random value, to make brute-force attacks unfeasible.

Specific Recommendations:

- Do not use the statistical pseudo-random number generator, use the cryptorandom generator instead. In Java, this is the SecureRandom class.

Source Code Examples

Java

Use of a weak pseudo-random number generator

```
Random random = new Random();

long sessNum = random.nextLong();

String sessionId = sessNum.toString();
```

```
```

Cryptographically secure random number generator

```
SecureRandom random = new SecureRandom();

byte sessBytes[] = new byte[32];

random.nextBytes(sessBytes);

String sessionId = new String(sessBytes);
```

Objc

Use of a weak pseudo-random number generator

```
long sessNum = rand();
NSString* sessionId = [NSString stringWithFormat:@"%ld", sessNum];
```

Cryptographically secure random number generator

```
UInt32 sessBytes;
SecRandomCopyBytes(kSecRandomDefault, sizeof(sessBytes), (uint8_t*)&sessBytes);

NSString* sessionId = [NSString stringWithFormat:@"%llu", sessBytes];
```

Swift

Use of a weak pseudo-random number generator

```
let sessNum = rand();
let sessionId = String(format:"%ld", sessNum)
```

Cryptographically secure random number generator

```
var sessBytes: UInt32 = 0
withUnsafeMutablePointer(&sessBytes, { (sessBytesPointer) -> Void in
    let castedPointer = unsafeBitCast(sessBytesPointer, UnsafeMutablePointer<UInt8>.self)
    SecRandomCopyBytes(kSecRandomDefault, sizeof(UInt32), castedPointer)
})

let sessionId = String(format:"%llu", sessBytes)
```

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';
```

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

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());
```

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 Dereference	**Development 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

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

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

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--;
       }
    }
}
```

Information Leak Through Comments

Weakness ID: 615 *(Weakness Variant)* **Status:** Incomplete

Description

Description Summary

While adding general comments is very useful, some programmers tend to leave important data, such as: filenames related to the web application, old links or links which were not meant to be browsed by users, old code fragments, etc.

Extended Description

An attacker who finds these comments can map the application's structure and files, expose hidden parts of the site, and study the fragments of code to reverse engineer the application, which may help develop further attacks against the site.

Time of Introduction

- Implementation

Demonstrative Examples

Example 1

The following comment, embedded in a JSP, will be displayed in the resulting HTML output.

(Bad Code)

Example Languages: **HTML and JSP**

```
<!-- FIXME: calling this with more than 30 args kills the JDBC server -->
```

Observed Examples

Reference	Description
CVE-2007-6197	Version numbers and internal hostnames leaked in HTML comments.
CVE-2007-4072	CMS places full pathname of server in HTML comment.
CVE-2009-2431	blog software leaks real username in HTML comment.

Potential Mitigations

Remove comments which have sensitive information about the design/implementation of the application. Some of the comments may be exposed to the user and affect the security posture of the application.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Variant	540	Information Leak Through Source Code	**Development Concepts (primary)699 Research Concepts (primary)1000**

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	Anonymous Tool Vendor (under NDA)		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Sean Eidemiller	Cigital	External
	added/updated demonstrative examples		
2008-07-01	Eric Dalci	Cigital	External
	updated Potential Mitigations, Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2008-10-14	CWE Content Team	MITRE	Internal
	updated Description		
2009-03-10	CWE Content Team	MITRE	Internal

	updated Demonstrative Examples		
2009-07-27	CWE Content Team	MITRE	Internal
	updated Observed Examples, Taxonomy Mappings		

BACK TO TOP

Scanned Languages

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