

## vul\_files\_75 Scan Report

Project Name	vul_files_75
Scan Start	Thursday, January 9, 2025 11:12:52 AM
Preset	Checkmarx Default
Scan Time	01h:23m:55s
Lines Of Code Scanned	297268
Files Scanned	58
Report Creation Time	Thursday, January 9, 2025 1:12:46 PM
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069</a>
Team	CxServer
Checkmarx Version	8.7.0
Scan Type	Full
Source Origin	LocalPath
Density	2/1000 (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**

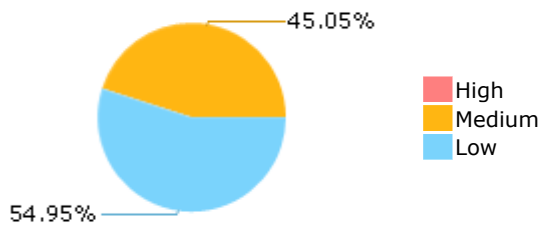
Results limit per query was set to 50

**Selected Queries**

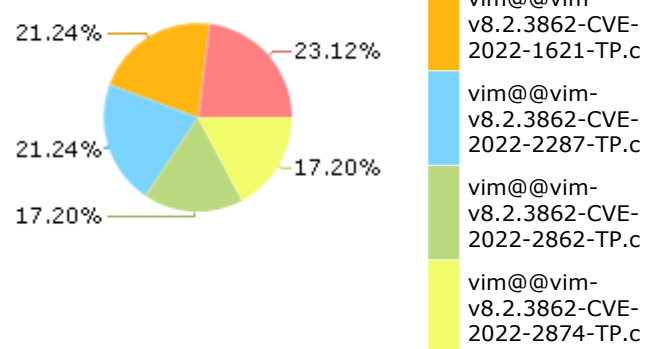
Selected queries are listed in [Result Summary](#)

---

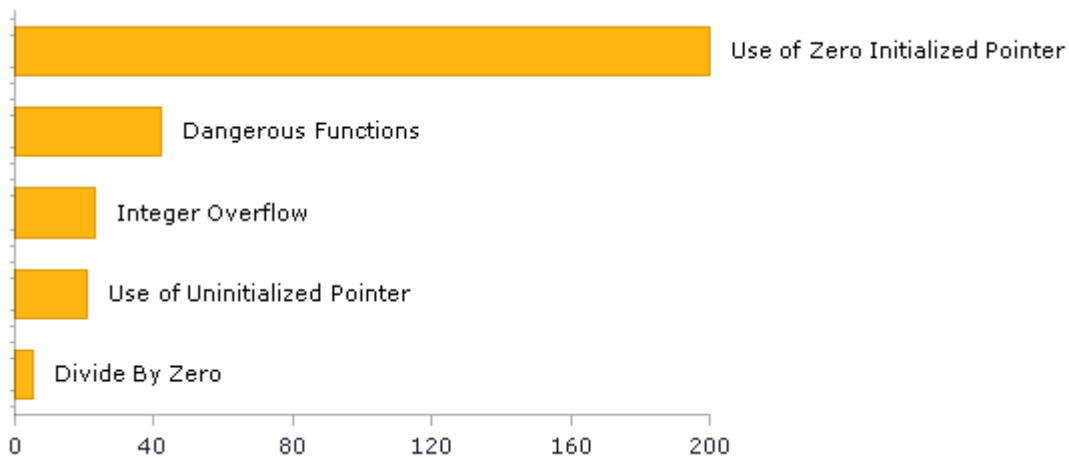
## Result Summary



## 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	194	60
A2-Broken Authentication	App. Specific	EASY	COMMON	AVERAGE	SEVERE	App. Specific	58	58
A3-Sensitive Data Exposure	App. Specific	AVERAGE	WIDESPREAD	AVERAGE	SEVERE	App. Specific	0	0
A4-XML External Entities (XXE)	App. Specific	AVERAGE	COMMON	EASY	SEVERE	App. Specific	0	0
A5-Broken Access Control*	App. Specific	AVERAGE	COMMON	AVERAGE	SEVERE	App. Specific	0	0
A6-Security Misconfiguration	App. Specific	EASY	WIDESPREAD	EASY	MODERATE	App. Specific	0	0
A7-Cross-Site Scripting (XSS)	App. Specific	EASY	WIDESPREAD	EASY	MODERATE	App. Specific	0	0
A8-Insecure Deserialization	App. Specific	DIFFICULT	COMMON	AVERAGE	SEVERE	App. Specific	0	0
A9-Using Components with Known Vulnerabilities*	App. Specific	AVERAGE	WIDESPREAD	AVERAGE	MODERATE	App. Specific	42	42
A10-Insufficient Logging & Monitoring	App. Specific	AVERAGE	WIDESPREAD	DIFFICULT	MODERATE	App. Specific	0	0

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

## Scan Summary - OWASP Top 10 2013

Further details and elaboration about vulnerabilities and risks can be found at: [OWASP Top 10 2013](#)

Category	Threat Agent	Attack Vectors	Weakness Prevalence	Weakness Detectability	Technical Impact	Business Impact	Issues Found	Best Fix Locations
A1-Injection	EXTERNAL, INTERNAL, ADMIN USERS	EASY	COMMON	AVERAGE	SEVERE	ALL DATA	0	0
A2-Broken Authentication and Session Management	EXTERNAL, INTERNAL USERS	AVERAGE	WIDESPREAD	AVERAGE	SEVERE	AFFECTED DATA AND FUNCTIONS	0	0
A3-Cross-Site Scripting (XSS)	EXTERNAL, INTERNAL, ADMIN USERS	AVERAGE	VERY WIDESPREAD	EASY	MODERATE	AFFECTED DATA AND SYSTEM	0	0
A4-Insecure Direct Object References	SYSTEM USERS	EASY	COMMON	EASY	MODERATE	EXPOSED DATA	0	0
A5-Security Misconfiguration	EXTERNAL, INTERNAL, ADMIN USERS	EASY	COMMON	EASY	MODERATE	ALL DATA AND SYSTEM	0	0
A6-Sensitive Data Exposure	EXTERNAL, INTERNAL, ADMIN USERS, USERS BROWSERS	DIFFICULT	UNCOMMON	AVERAGE	SEVERE	EXPOSED DATA	0	0
A7-Missing Function Level Access Control*	EXTERNAL, INTERNAL USERS	EASY	COMMON	AVERAGE	MODERATE	EXPOSED DATA AND FUNCTIONS	0	0
A8-Cross-Site Request Forgery (CSRF)	USERS BROWSERS	AVERAGE	COMMON	EASY	MODERATE	AFFECTED DATA AND FUNCTIONS	0	0
A9-Using Components with Known Vulnerabilities*	EXTERNAL USERS, AUTOMATED TOOLS	AVERAGE	WIDESPREAD	DIFFICULT	MODERATE	AFFECTED DATA AND FUNCTIONS	42	42
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	5	5
PCI DSS (3.2) - 6.5.2 - Buffer overflows	25	25
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.	1	1
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.	7	7
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.	57	57
Media Protection	Organizations must: (i) protect information system media, both paper and digital; (ii) limit access to information on information system media to authorized users; and (iii) sanitize or destroy information system media before disposal or release for reuse.	0	0
System And Communications Protection	Organizations must: (i) monitor, control, and protect organizational communications (i.e., information transmitted or received by organizational information systems) at the external boundaries and key internal boundaries of the information systems; and (ii) employ architectural designs, software development techniques, and systems engineering principles that promote effective information security within organizational information systems.	0	0
System And Information Integrity	Organizations must: (i) identify, report, and correct information and information system flaws in a timely manner; (ii) provide protection from malicious code at appropriate locations within organizational information systems; and (iii) monitor information system security alerts and advisories and take appropriate actions in response.	23	23

\* 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)	58	58
AC-4 Information Flow Enforcement (P1)	0	0
AC-6 Least Privilege (P1)	0	0
AU-9 Protection of Audit Information (P1)	0	0
CM-6 Configuration Settings (P2)	0	0
IA-5 Authenticator Management (P1)	0	0
IA-6 Authenticator Feedback (P2)	0	0
IA-8 Identification and Authentication (Non-Organizational Users) (P1)	0	0
SC-12 Cryptographic Key Establishment and Management (P1)	0	0
SC-13 Cryptographic Protection (P1)	0	0
SC-17 Public Key Infrastructure Certificates (P1)	0	0
SC-18 Mobile Code (P2)	0	0
SC-23 Session Authenticity (P1)*	0	0
SC-28 Protection of Information at Rest (P1)	0	0
SC-4 Information in Shared Resources (P1)	0	0
SC-5 Denial of Service Protection (P1)*	402	114
SC-8 Transmission Confidentiality and Integrity (P1)	0	0
SI-10 Information Input Validation (P1)*	69	69
SI-11 Error Handling (P2)*	25	25
SI-15 Information Output Filtering (P0)	0	0
SI-16 Memory Protection (P1)	5	5

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



## Scan Summary - 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 wasn't 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

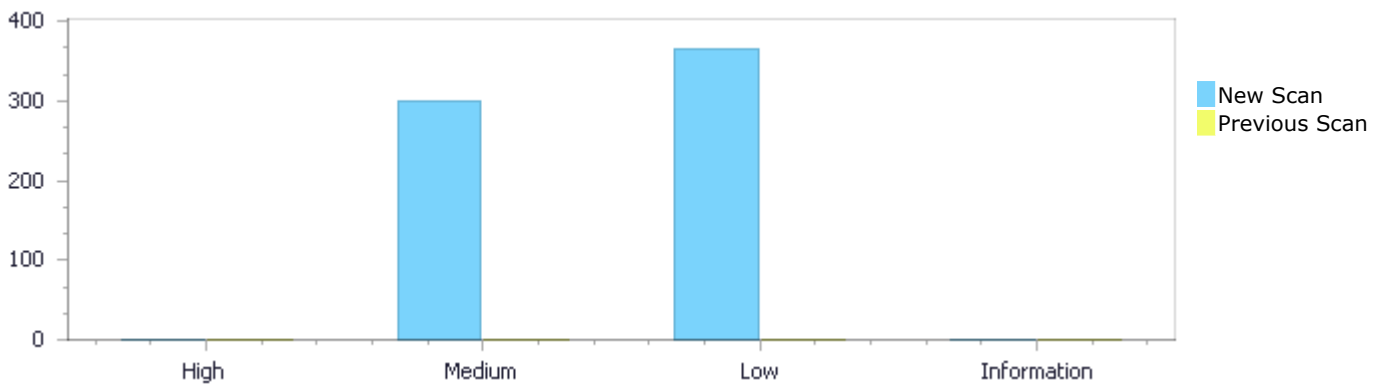
Category	Issues Found	Best Fix Locations
Must audit	0	0
Check	0	0
Optional	0	0

## Results Distribution By Status

First scan of the project

	High	Medium	Low	Information	Total
New Issues	0	300	366	0	666
Recurrent Issues	0	0	0	0	0
Total	0	300	366	0	666

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	300	366	0	666
Urgent	0	0	0	0	0
Proposed Not Exploitable	0	0	0	0	0
Total	0	300	366	0	666

## Result Summary

Vulnerability Type	Occurrences	Severity
<a href="#">Use of Zero Initialized Pointer</a>	200	Medium
<a href="#">Dangerous Functions</a>	42	Medium
<a href="#">Integer Overflow</a>	23	Medium
<a href="#">Use of Uninitialized Pointer</a>	21	Medium
<a href="#">Divide By Zero</a>	5	Medium

<a href="#">MemoryFree on StackVariable</a>	4	Medium
<a href="#">Use of Uninitialized Variable</a>	2	Medium
<a href="#">Buffer Overflow AddressOfLocalVarReturned</a>	1	Medium
<a href="#">Char Overflow</a>	1	Medium
<a href="#">Memory Leak</a>	1	Medium
<a href="#">NULL Pointer Dereference</a>	170	Low
<a href="#">Improper Resource Access Authorization</a>	57	Low
<a href="#">Use of Sizeof On a Pointer Type</a>	55	Low
<a href="#">Unchecked Array Index</a>	27	Low
<a href="#">Unchecked Return Value</a>	25	Low
<a href="#">Potential Precision Problem</a>	18	Low
<a href="#">Arithmenic Operation On Boolean</a>	7	Low
<a href="#">Potential Off by One Error in Loops</a>	5	Low
<a href="#">Incorrect Permission Assignment For Critical Resources</a>	1	Low
<a href="#">TOCTOU</a>	1	Low

## 10 Most Vulnerable Files

### High and Medium Vulnerabilities

File Name	Issues Found
vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	59
vim@@vim-v8.2.3862-CVE-2022-2874-TP.c	59
vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	35
vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	19
vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	19
vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	12
vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	10
vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	10
vim@@vim-v8.2.3862-CVE-2022-2125-TP.c	9
vim@@vim-v8.2.3862-CVE-2022-2183-TP.c	9

## Scan Results Details

### Use of Zero Initialized Pointer

Query Path:

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

#### Categories

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

#### Description

##### Use of Zero Initialized Pointer\Path 1:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=101">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=101</a>
Status	New

The variable declared in ht at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 816 is not initialized when it is used by ll\_tv at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 816.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	834	933
Object	ht	ll_tv

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method get\_lval(

```

.....
834.      hashtable_T      *ht = NULL;
.....
933.      lp->ll_tv = &v->di_tv;

```

##### Use of Zero Initialized Pointer\Path 2:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=102">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=102</a>
Status	New

The variable declared in ht\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4681 is not initialized when it is used by ht\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4681.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-	vim@@vim-v8.2.3862-CVE-2022-1420-

	FP.c	FP.c
Line	4687	4714
Object	ht_stack	ht_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_ht(hashtab\_T \*ht, int copyID, list\_stack\_T \*\*list\_stack)

```
....  
4687.      ht_stack_T    *ht_stack = NULL;  
....  
4714.      ht_stack = ht_stack->prev;
```

#### Use of Zero Initialized Pointer\Path 3:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=103">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=103</a>
Status	New

The variable declared in ht\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4681 is not initialized when it is used by ht\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4681.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4687	4712
Object	ht_stack	ht_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_ht(hashtab\_T \*ht, int copyID, list\_stack\_T \*\*list\_stack)

```
....  
4687.      ht_stack_T    *ht_stack = NULL;  
....  
4712.      cur_ht = ht_stack->ht;
```

#### Use of Zero Initialized Pointer\Path 4:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=104">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=104</a>
Status	New

The variable declared in list\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4758 is not initialized when it is used by list\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4758.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4763	4782
Object	list_stack	list_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_list\_items(list\_T \*l, int copyID, ht\_stack\_T \*\*ht\_stack)

```
....  
4763.      list_stack_T *list_stack = NULL;  
....  
4782.      list_stack = list_stack->prev;
```

#### Use of Zero Initialized Pointer\Path 5:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=105">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=105</a>
Status	New

The variable declared in list\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4758 is not initialized when it is used by list\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4758.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4763	4780
Object	list_stack	list_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_list\_items(list\_T \*l, int copyID, ht\_stack\_T \*\*ht\_stack)

```
....  
4763.      list_stack_T *list_stack = NULL;  
....  
4780.      cur_l = list_stack->list;
```

#### Use of Zero Initialized Pointer\Path 6:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=106">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=106</a>
Status	New



The variable declared in saved\_line at vim@@vim-v8.2.3862-CVE-2022-1735-TP.c in line 4845 is not initialized when it is used by saved\_line at vim@@vim-v8.2.3862-CVE-2022-1735-TP.c in line 4845.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1735-TP.c	vim@@vim-v8.2.3862-CVE-2022-1735-TP.c
Line	4946	5098
Object	saved_line	saved_line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1735-TP.c

Method ins\_tab(void)

```

....
4946.      char_u      *saved_line = NULL;      // init for GCC
....
5098.      ins_bytes_len(saved_line + change_col,
```

#### Use of Zero Initialized Pointer\Path 7:

Severity Medium

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=107>

Status New

The variable declared in next\_leader\_flags at vim@@vim-v8.2.3862-CVE-2022-1851-TP.c in line 938 is not initialized when it is used by flags at vim@@vim-v8.2.3862-CVE-2022-1851-TP.c in line 461.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c
Line	1007	479
Object	next_leader_flags	flags

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1851-TP.c

Method format\_lines(

```

....
1007.      next_leader_flags = NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-1851-TP.c

Method fmt\_check\_par(

```

....
479.      flags = *leader_flags;
```

**Use of Zero Initialized Pointer\Path 8:**

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=108">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=108</a>
Status	New

The variable declared in leader\_flags at vim@@vim-v8.2.3862-CVE-2022-1851-TP.c in line 938 is not initialized when it is used by flags at vim@@vim-v8.2.3862-CVE-2022-1851-TP.c in line 461.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c
Line	950	479
Object	leader_flags	flags

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-1851-TP.c  
Method format\_lines(

```
....  
950.      char_u  *leader_flags = NULL;    // flags for leader of current  
line
```

File Name vim@@vim-v8.2.3862-CVE-2022-1851-TP.c  
Method fmt\_check\_par(

```
....  
479.      flags = *leader_flags;
```

**Use of Zero Initialized Pointer\Path 9:**

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=109">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=109</a>
Status	New

The variable declared in new\_vts\_array at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 1574 is not initialized when it is used by new\_line at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 1574.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c
Line	1590	1703
Object	new_vts_array	new_line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c  
Method ex\_retab(exarg\_T \*eap)

```
....
1590.      int      *new_vts_array = NULL;
....
1703.      new_line = alloc(old_len - col + start_col + len
+ 1);
```

#### Use of Zero Initialized Pointer\Path 10:

Severity Medium  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=110>  
Status New

The variable declared in Pointer at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 24 is not initialized when it is used by new\_line at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 1574.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c
Line	78	1703
Object	Pointer	new_line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c  
Method tabstop\_set(char\_u \*var, int \*\*array)

```
....
78.      *array = NULL;
```



File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c  
Method ex\_retab(exarg\_T \*eap)

```
....
1703.      new_line = alloc(old_len - col + start_col + len
+ 1);
```

#### Use of Zero Initialized Pointer\Path 11:

Severity Medium  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=111>  
Status New

The variable declared in Pointer at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 24 is not initialized when it is used by new\_line at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 1574.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c
Line	32	1703
Object	Pointer	new_line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c  
Method tabstop\_set(char\_u \*var, int \*\*array)

```
....  
32.    *array = NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c  
Method ex\_retab(exarg\_T \*eap)

```
....  
1703.                                     new_line = alloc(old_len - col + start_col + len  
+ 1);
```

#### Use of Zero Initialized Pointer\Path 12:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=112">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=112</a>
Status	New

The variable declared in new\_vts\_array at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 1574 is not initialized when it is used by new\_line at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 1574.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c
Line	1590	1710
Object	new_vts_array	new_line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c  
Method ex\_retab(exarg\_T \*eap)

```

.....
1590.      int      *new_vts_array = NULL;
.....
1710.                  ptr = new_line + start_col;

```

### Use of Zero Initialized Pointer\Path 13:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=113">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=113</a>
Status	New

The variable declared in Pointer at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 24 is not initialized when it is used by new\_line at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 1574.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c
Line	78	1710
Object	Pointer	new_line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c  
Method tabstop\_set(char\_u \*var, int \*\*array)

```

.....
78.      *array = NULL;

```

File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c  
Method ex\_retab(exarg\_T \*eap)

```

.....
1710.                  ptr = new_line + start_col;

```

### Use of Zero Initialized Pointer\Path 14:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=114">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=114</a>
Status	New

The variable declared in Pointer at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 24 is not initialized when it is used by new\_line at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 1574.

Source	Destination
--------	-------------

File	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c
Line	32	1710
Object	Pointer	new_line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c  
Method tabstop\_set(char\_u \*var, int \*\*array)

```
....
32.      *array = NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c  
Method ex\_retab(exarg\_T \*eap)

```
....
1710.                                     ptr = new_line + start_col;
```

#### Use of Zero Initialized Pointer\Path 15:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=115">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=115</a>
Status	New

The variable declared in new\_vts\_array at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 1574 is not initialized when it is used by new\_line at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 1574.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c
Line	1590	1703
Object	new_vts_array	new_line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method ex\_retab(exarg\_T \*eap)

```
....
1590.      int      *new_vts_array = NULL;
....
1703.      new_line = alloc(old_len - col + start_col + len
+ 1);
```

#### Use of Zero Initialized Pointer\Path 16:

Severity	Medium
Result State	To Verify

Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=116">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=116</a>
Status	New

The variable declared in Pointer at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 24 is not initialized when it is used by new\_line at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 1574.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c
Line	32	1703
Object	Pointer	new_line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method tabstop\_set(char\_u \*var, int \*\*array)

```
....
32.    *array = NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method ex\_retab(exarg\_T \*eap)

```
....
1703.                                new_line = alloc(old_len - col + start_col + len
+ 1);
```

#### Use of Zero Initialized Pointer\Path 17:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=117">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=117</a>
Status	New

The variable declared in Pointer at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 24 is not initialized when it is used by new\_line at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 1574.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c
Line	78	1703
Object	Pointer	new_line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method tabstop\_set(char\_u \*var, int \*\*array)

```
....
78.      *array = NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method ex\_retab(exarg\_T \*eap)

```
....
1703.      new_line = alloc(old_len - col + start_col + len
+ 1);
```

#### Use of Zero Initialized Pointer\Path 18:

Severity Medium  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=118>  
Status New

The variable declared in new\_vts\_array at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 1574 is not initialized when it is used by new\_line at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 1574.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c
Line	1590	1710
Object	new_vts_array	new_line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method ex\_retab(exarg\_T \*eap)

```
....
1590.      int      *new_vts_array = NULL;
....
1710.      ptr = new_line + start_col;
```

#### Use of Zero Initialized Pointer\Path 19:

Severity Medium  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=119>  
Status New

The variable declared in Pointer at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 24 is not initialized when it is used by new\_line at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 1574.

Source	Destination
--------	-------------



File	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c
Line	32	1710
Object	Pointer	new_line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method tabstop\_set(char\_u \*var, int \*\*array)

```
....
32.    *array = NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method ex\_retab(exarg\_T \*eap)

```
....
1710.                                ptr = new_line + start_col;
```

#### Use of Zero Initialized Pointer\Path 20:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=120">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=120</a>
Status	New

The variable declared in Pointer at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 24 is not initialized when it is used by new\_line at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 1574.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c
Line	78	1710
Object	Pointer	new_line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method tabstop\_set(char\_u \*var, int \*\*array)

```
....
78.    *array = NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method ex\_retab(exarg\_T \*eap)

```
.....
1710.                                ptr = new_line + start_col;
```

### Use of Zero Initialized Pointer\Path 21:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=121">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=121</a>
Status	New

The variable declared in saved\_line at vim@@vim-v8.2.3862-CVE-2022-2207-TP.c in line 4845 is not initialized when it is used by saved\_line at vim@@vim-v8.2.3862-CVE-2022-2207-TP.c in line 4845.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2207-TP.c	vim@@vim-v8.2.3862-CVE-2022-2207-TP.c
Line	4946	5098
Object	saved_line	saved_line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2207-TP.c  
Method ins\_tab(void)

```
.....
4946.          char_u          *saved_line = NULL;          // init for GCC
.....
5098.          ins_bytes_len(saved_line + change_col,
```

### Use of Zero Initialized Pointer\Path 22:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=122">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=122</a>
Status	New

The variable declared in ht at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 816 is not initialized when it is used by ll\_tv at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 816.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	834	933
Object	ht	ll_tv

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c

Method get\_lval(

```
....  
834.      hashtable_T      *ht = NULL;  
....  
933.      lp->ll_tv = &v->di_tv;
```

### Use of Zero Initialized Pointer\Path 23:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=123">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=123</a>
Status	New

The variable declared in ht\_stack at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 4681 is not initialized when it is used by ht\_stack at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 4681.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	4687	4714
Object	ht_stack	ht_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method set\_ref\_in\_ht(hashtable\_T \*ht, int copyID, list\_stack\_T \*\*list\_stack)

```
....  
4687.      ht_stack_T      *ht_stack = NULL;  
....  
4714.      ht_stack = ht_stack->prev;
```

### Use of Zero Initialized Pointer\Path 24:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=124">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=124</a>
Status	New

The variable declared in ht\_stack at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 4681 is not initialized when it is used by ht\_stack at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 4681.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	4687	4712
Object	ht_stack	ht_stack

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method set\_ref\_in\_ht(hashtab\_T \*ht, int copyID, list\_stack\_T \*\*list\_stack)

```
....  
4687.      ht_stack_T      *ht_stack = NULL;  
....  
4712.      cur_ht = ht_stack->ht;
```

**Use of Zero Initialized Pointer\Path 25:**

Severity Medium  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=125>  
Status New

The variable declared in list\_stack at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 4758 is not initialized when it is used by list\_stack at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 4758.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	4763	4782
Object	list_stack	list_stack

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method set\_ref\_in\_list\_items(list\_T \*l, int copyID, ht\_stack\_T \*\*ht\_stack)

```
....  
4763.      list_stack_T *list_stack = NULL;  
....  
4782.      list_stack = list_stack->prev;
```

**Use of Zero Initialized Pointer\Path 26:**

Severity Medium  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=126>  
Status New

The variable declared in list\_stack at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 4758 is not initialized when it is used by list\_stack at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 4758.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	4763	4780

Object	list_stack	list_stack
--------	------------	------------

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method set\_ref\_in\_list\_items(list\_T \*l, int copyID, ht\_stack\_T \*\*ht\_stack)

```
....
4763.      list_stack_T *list_stack = NULL;
....
4780.      cur_l = list_stack->list;
```

#### Use of Zero Initialized Pointer\Path 27:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=127">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=127</a>
Status	New

The variable declared in saved\_line at vim@@vim-v8.2.3862-CVE-2022-2845-TP.c in line 4845 is not initialized when it is used by saved\_line at vim@@vim-v8.2.3862-CVE-2022-2845-TP.c in line 4845.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2845-TP.c	vim@@vim-v8.2.3862-CVE-2022-2845-TP.c
Line	4946	5098
Object	saved_line	saved_line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2845-TP.c  
Method ins\_tab(void)

```
....
4946.      char_u      *saved_line = NULL;      // init for GCC
....
5098.      ins_bytes_len(saved_line + change_col,
```

#### Use of Zero Initialized Pointer\Path 28:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=128">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=128</a>
Status	New

The variable declared in line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-	vim@@vim-v8.2.3862-CVE-2022-2862-

	TP.c	TP.c
Line	2885	3044
Object	line	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....
2885.             line = NULL;
....
3044.             line = skipwhite(line);
```

#### Use of Zero Initialized Pointer\Path 29:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=129">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=129</a>
Status	New

The variable declared in line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	2874	3044
Object	line	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....
2874.             line = NULL;
....
3044.             line = skipwhite(line);
```

#### Use of Zero Initialized Pointer\Path 30:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=130">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=130</a>
Status	New

The variable declared in line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	2428	3044
Object	line	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....
2428.      char_u *line = NULL;
....
3044.      line = skipwhite(line);
```

### Use of Zero Initialized Pointer\Path 31:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=131">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=131</a>
Status	New

The variable declared in line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422 is not initialized when it is used by ga\_data at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	2428	2543
Object	line	ga_data

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....
2428.      char_u *line = NULL;
....
2543.      val_type = ((type_T **)stack->ga_data)[stack->ga_len - 1];
```

### Use of Zero Initialized Pointer\Path 32:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=132">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=132</a>
Status	New

The variable declared in != at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 3173 is not initialized when it is used by ga\_data at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	3175	2543
Object	!=	ga_data

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method set\_function\_type(ufunc\_T \*ufunc)

```
....  
3175.         int varargs = ufunc->uf_va_name != NULL;
```



File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....  
2543.         val_type = ((type_T **) stack->ga_data) [stack->ga_len -  
1];
```

#### Use of Zero Initialized Pointer\Path 33:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=133">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=133</a>
Status	New

The variable declared in line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	2428	2859
Object	line	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(



```

....
2428.         char_u *line = NULL;
....
2859.                 line = compile_return(p, check_return_type,

```

#### Use of Zero Initialized Pointer\Path 34:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=134">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=134</a>
Status	New

The variable declared in != at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 3173 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	3175	2859
Object	!=	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method set\_function\_type(ufunc\_T \*ufunc)

```

....
3175.         int varargs = ufunc->uf_va_name != NULL;

```

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```

....
2859.                 line = compile_return(p, check_return_type,

```

#### Use of Zero Initialized Pointer\Path 35:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=135">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=135</a>
Status	New

The variable declared in line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

Source	Destination
--------	-------------

File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	2428	2970
Object	line	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....
2428.      char_u *line = NULL;
....
2970.                  line = compile_substitute(line, &ea, &cctx);
```

#### Use of Zero Initialized Pointer\Path 36:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=136">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=136</a>
Status	New

The variable declared in != at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 3173 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	3175	2970
Object	!=	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method set\_function\_type(ufunc\_T \*ufunc)

```
....
3175.      int varargs = ufunc->uf_va_name != NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....
2970.                  line = compile_substitute(line, &ea, &cctx);
```

#### Use of Zero Initialized Pointer\Path 37:

Severity	Medium
Result State	To Verify

Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=137">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=137</a>
Status	New

The variable declared in line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	2428	2976
Object	line	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....  
2428.      char_u *line = NULL;  
....  
2976.          line = compile_redir(line, &ea, &cctx);
```

#### Use of Zero Initialized Pointer\Path 38:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=138">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=138</a>
Status	New

The variable declared in != at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 3173 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	3175	2976
Object	!=	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method set\_function\_type(ufunc\_T \*ufunc)

```
....  
3175.      int varargs = ufunc->uf_va_name != NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c

Method compile\_def\_function(

```
....
2976.                line = compile_redir(line, &ea, &cctx);
```

### Use of Zero Initialized Pointer\Path 39:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=139">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=139</a>
Status	New

The variable declared in line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	2428	2987
Object	line	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....
2428.        char_u *line = NULL;
....
2987.                line = compile_cexpr(line, &ea, &cctx);
```

### Use of Zero Initialized Pointer\Path 40:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=140">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=140</a>
Status	New

The variable declared in != at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 3173 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	3175	2987
Object	!=	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method set\_function\_type(ufunc\_T \*ufunc)

```
....  
3175.          int varargs = ufunc->uf_va_name != NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....  
2987.          line = compile_cexpr(line, &ea, &cctx);
```

#### Use of Zero Initialized Pointer\Path 41:

Severity Medium  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=141>  
Status New

The variable declared in line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	2428	3038
Object	line	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....  
2428.          char_u *line = NULL;  
....  
3038.          line = compile_exec(line, &ea, &cctx);
```

#### Use of Zero Initialized Pointer\Path 42:

Severity Medium  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=142>  
Status New

The variable declared in != at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 3173 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	3175	3038
Object	!=	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method set\_function\_type(ufunc\_T \*ufunc)

```
....
3175.         int varargs = ufunc->uf_va_name != NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....
3038.         line = compile_exec(line, &ea, &cctx);
```

#### Use of Zero Initialized Pointer\Path 43:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=143">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=143</a>
Status	New

The variable declared in line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	2428	2811
Object	line	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....
2428.         char_u *line = NULL;
....
2811.         line += STRLEN(line);
```

#### Use of Zero Initialized Pointer\Path 44:

Severity	Medium
----------	--------

Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=144">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=144</a>
Status	New

The variable declared in line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	2428	2880
Object	line	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....
2428.      char_u *line = NULL;
....
2880.                  line = compile_unletlock(p, &ea, &cctx);
```

#### Use of Zero Initialized Pointer\Path 45:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=145">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=145</a>
Status	New

The variable declared in != at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 3173 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	3175	2880
Object	!=	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method set\_function\_type(ufunc\_T \*ufunc)

```
....
3175.      int varargs = ufunc->uf_va_name != NULL;
```



File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c

Method compile\_def\_function(

```
....  
2880. line = compile_unletlock(p, &ea, &cctx);
```

#### Use of Zero Initialized Pointer\Path 46:

Severity Medium

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=146>

Status New

The variable declared in line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	2428	2959
Object	line	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c

Method compile\_def\_function(

```
....  
2428. char_u *line = NULL;  
....  
2959. line = compile_put(p, &ea, &cctx);
```

#### Use of Zero Initialized Pointer\Path 47:

Severity Medium

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=147>

Status New

The variable declared in != at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 3173 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	3175	2959
Object	!=	line



#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method set\_function\_type(ufunc\_T \*ufunc)

```
....
3175.          int varargs = ufunc->uf_va_name != NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....
2959.          line = compile_put(p, &ea, &cctx);
```

#### Use of Zero Initialized Pointer\Path 48:

Severity Medium  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=148>  
Status New

The variable declared in line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	2428	2889
Object	line	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....
2428.          char_u *line = NULL;
....
2889.          line = compile_if(p, &cctx);
```

#### Use of Zero Initialized Pointer\Path 49:

Severity Medium  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=149>  
Status New

The variable declared in != at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 3173 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	3175	2889
Object	!=	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method set\_function\_type(ufunc\_T \*ufunc)

```
....
3175.          int varargs = ufunc->uf_va_name != NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....
2889.          line = compile_if(p, &cctx);
```

#### Use of Zero Initialized Pointer\Path 50:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=150">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=150</a>
Status	New

The variable declared in line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2422 is not initialized when it is used by line at vim@@vim-v8.2.3862-CVE-2022-2862-TP.c in line 2892.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	2428	2892
Object	line	line

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....
2428.          char_u *line = NULL;
....
2892.          line = compile_elseif(p, &cctx);
```

## Dangerous Functions

Query Path:

## 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	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=35">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=35</a>
Status	New

The dangerous function, printf, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-1621-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	2746	2746
Object	printf	printf

### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
....
2746.                                printf((char *)buf, "%s", items[4]);
```

### Dangerous Functions\Path 2:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=36">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=36</a>
Status	New

The dangerous function, printf, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-1621-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	2748	2748
Object	printf	printf

### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c

Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
....  
2748.                                sprintf((char *)buf, "%s$", items[4]);
```

### Dangerous Functions\Path 3:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=37">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=37</a>
Status	New

The dangerous function, sprintf, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-1621-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	2807	2807
Object	sprintf	sprintf

### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
....  
2807.                                sprintf((char *)buf, "^%s",
```

### Dangerous Functions\Path 4:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=38">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=38</a>
Status	New

The dangerous function, sprintf, was found in use at line 1242 in vim@@vim-v8.2.3862-CVE-2022-2126-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	1370	1370
Object	sprintf	sprintf

### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
.....
1370.                                sprintf(changename[depth], "prefix");
```

### Dangerous Functions\Path 5:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=39">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=39</a>
Status	New

The dangerous function, sprintf, was found in use at line 1242 in vim@@vim-v8.2.3862-CVE-2022-2126-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	1769	1769
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
.....
1769.                                sprintf(changename[depth], "%.s-%s: split",
```

### Dangerous Functions\Path 6:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=40">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=40</a>
Status	New

The dangerous function, sprintf, was found in use at line 1242 in vim@@vim-v8.2.3862-CVE-2022-2126-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	1772	1772
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....
1772.                                sprintf(changename[depth], "%.s-%s:
compound",
```

### Dangerous Functions\Path 7:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=41">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=41</a>
Status	New

The dangerous function, sprintf, was found in use at line 1242 in vim@@vim-v8.2.3862-CVE-2022-2126-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	1918	1918
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....
1918.                                sprintf(changename[depth], "%.s-%s: subst %c to
%c",
```

### Dangerous Functions\Path 8:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=42">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=42</a>
Status	New

The dangerous function, sprintf, was found in use at line 1242 in vim@@vim-v8.2.3862-CVE-2022-2126-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	1922	1922
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c

Method suggest\_trie\_walk(

```
....  
1922.                sprintf(changename[depth], "%.s-%s: accept %c",
```

### Dangerous Functions\Path 9:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=43">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=43</a>
Status	New

The dangerous function, sprintf, was found in use at line 1242 in vim@@vim-v8.2.3862-CVE-2022-2126-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2057	2057
Object	sprintf	sprintf

### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....  
2057.                sprintf(changename[depth], "%.s-%s: delete %c",
```

### Dangerous Functions\Path 10:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=44">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=44</a>
Status	New

The dangerous function, sprintf, was found in use at line 1242 in vim@@vim-v8.2.3862-CVE-2022-2126-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2154	2154
Object	sprintf	sprintf

### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
.....
2154.                sprintf(changename[depth], "%.s-%s: insert %c",
```

### Dangerous Functions\Path 11:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=45">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=45</a>
Status	New

The dangerous function, sprintf, was found in use at line 1242 in vim@@vim-v8.2.3862-CVE-2022-2126-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2254	2254
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
.....
2254.                sprintf(changename[depth], "%.s-%s: swap %c and %c",
```

### Dangerous Functions\Path 12:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=46">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=46</a>
Status	New

The dangerous function, sprintf, was found in use at line 1242 in vim@@vim-v8.2.3862-CVE-2022-2126-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2342	2342
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(



```
.....
2342.                sprintf(changename[depth], "%.s-%s: swap3 %c and %c",
```

### Dangerous Functions\Path 13:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=47">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=47</a>
Status	New

The dangerous function, sprintf, was found in use at line 1242 in vim@@vim-v8.2.3862-CVE-2022-2126-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2410	2410
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
.....
2410.                sprintf(changename[depth], "%.s-%s: rotate left
%c%c%c",
```

### Dangerous Functions\Path 14:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=48">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=48</a>
Status	New

The dangerous function, sprintf, was found in use at line 1242 in vim@@vim-v8.2.3862-CVE-2022-2126-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2471	2471
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
.....
2471.                sprintf(changename[depth], "%.s-%s: rotate right
%c%c%c",
```

### Dangerous Functions\Path 15:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=49">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=49</a>
Status	New

The dangerous function, sprintf, was found in use at line 1242 in vim@@vim-v8.2.3862-CVE-2022-2126-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2583	2583
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
.....
2583.                sprintf(changename[depth], "%.s-%s: replace %s
with %s",
```

### Dangerous Functions\Path 16:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=50">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=50</a>
Status	New

The dangerous function, sprintf, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-2287-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	2746	2746
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c

Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
....  
2746.                                sprintf((char *)buf, "^%s", items[4]);
```

### Dangerous Functions\Path 17:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=51">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=51</a>
Status	New

The dangerous function, sprintf, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-2287-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	2748	2748
Object	sprintf	sprintf

### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
....  
2748.                                sprintf((char *)buf, "%s$", items[4]);
```

### Dangerous Functions\Path 18:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=52">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=52</a>
Status	New

The dangerous function, sprintf, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-2287-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	2807	2807
Object	sprintf	sprintf

### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
.....  
2807.                                sprintf((char *)buf, "^%s",
```

### Dangerous Functions\Path 19:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=53">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=53</a>
Status	New

The dangerous function, sprintf, was found in use at line 722 in vim@@vim-v8.2.3862-CVE-2022-2816-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2816-TP.c	vim@@vim-v8.2.3862-CVE-2022-2816-TP.c
Line	762	762
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2816-TP.c  
Method ex\_compiler(exarg\_T \*eap)

```
.....  
762.                                sprintf((char *)buf, "compiler/%s.vim", eap->arg);
```

### Dangerous Functions\Path 20:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=54">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=54</a>
Status	New

The dangerous function, sprintf, was found in use at line 499 in vim@@vim-v8.2.3862-CVE-2022-2889-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	504	504
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c  
Method get\_lambda\_name(void)

```
....  
504.      sprintf((char*)name, "<lambda>%d", ++lambda_no);
```

### Dangerous Functions\Path 21:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=55">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=55</a>
Status	New

The dangerous function, sprintf, was found in use at line 1813 in vim@@vim-v8.2.3862-CVE-2022-2889-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	1832	1832
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c  
Method fname\_trans\_sid(char\_u \*name, char\_u \*fname\_buf, char\_u \*\*tofree, int \*error)

```
....  
1832.      sprintf((char *)fname_buf + 3, "%ld_",
```

### Dangerous Functions\Path 22:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=56">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=56</a>
Status	New

The dangerous function, sprintf, was found in use at line 2456 in vim@@vim-v8.2.3862-CVE-2022-2889-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	2642	2642
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c  
Method call\_user\_func(

```
.....  
2642.          sprintf((char *)numbuf, "%d", ai + 1);
```

### Dangerous Functions\Path 23:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=57">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=57</a>
Status	New

The dangerous function, sprintf, was found in use at line 2966 in vim@@vim-v8.2.3862-CVE-2022-2889-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	2977	2977
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c  
Method delete\_script\_functions(int sid)

```
.....  
2977.          sprintf((char *)buf + 3, "%d_", sid);
```

### Dangerous Functions\Path 24:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=58">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=58</a>
Status	New

The dangerous function, sprintf, was found in use at line 3607 in vim@@vim-v8.2.3862-CVE-2022-2889-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	3811	3811
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c  
Method trans\_function\_name(

```
....  
3811.          sprintf((char *)sid_buf, "%ld_",  
(long)current_sctx.sc_sid);
```

### Dangerous Functions\Path 25:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=59">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=59</a>
Status	New

The dangerous function, sprintf, was found in use at line 3951 in vim@@vim-v8.2.3862-CVE-2022-2889-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	4434	4434
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c  
Method define\_function(exarg\_T \*eap, char\_u \*name\_arg)

```
....  
4434.          sprintf(numbuf, "%d", ++func_nr);
```

### Dangerous Functions\Path 26:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=60">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=60</a>
Status	New

The dangerous function, strlen, was found in use at line 707 in vim@@vim-v8.2.3862-CVE-2022-1733-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1733-TP.c	vim@@vim-v8.2.3862-CVE-2022-1733-TP.c
Line	723	723
Object	strlen	strlen

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1733-TP.c  
Method cin\_isinit(void)

```
.....
723.                l = (int)strlen(skip[i]);
```

### Dangerous Functions\Path 27:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=61">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=61</a>
Status	New

The dangerous function, atoi, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-1621-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	2473	2473
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
.....
2473.                if (atoi((char *)items[1]) == 0)
```

### Dangerous Functions\Path 28:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=62">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=62</a>
Status	New

The dangerous function, atoi, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-1621-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	2503	2503
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)



```
.....  
2503.                compmax = atoi((char *)items[1]);
```

### Dangerous Functions\Path 29:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=63">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=63</a>
Status	New

The dangerous function, atoi, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-1621-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	2511	2511
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
.....  
2511.                compminlen = atoi((char *)items[1]);
```

### Dangerous Functions\Path 30:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=64">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=64</a>
Status	New

The dangerous function, atoi, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-1621-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	2519	2519
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
.....  
2519.                compsymmax = atoi((char *)items[1]);
```

### Dangerous Functions\Path 31:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=65">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=65</a>
Status	New

The dangerous function, atoi, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-1621-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	2542	2542
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
.....  
2542.                if (atoi((char *)items[1]) == 0)
```

### Dangerous Functions\Path 32:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=66">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=66</a>
Status	New

The dangerous function, atoi, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-1621-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	2688	2688
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
.....
2688.                aff_todo = atoi((char *)items[3]);
```

### Dangerous Functions\Path 33:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=67">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=67</a>
Status	New

The dangerous function, atoi, was found in use at line 2806 in vim@@vim-v8.2.3862-CVE-2022-1886-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c
Line	2835	2835
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1886-TP.c  
Method write\_reg\_contents\_ex(

```
.....
2835.                int    num = atoi((char *)str);
```

### Dangerous Functions\Path 34:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=68">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=68</a>
Status	New

The dangerous function, atoi, was found in use at line 24 in vim@@vim-v8.2.3862-CVE-2022-2125-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c
Line	71	71
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c  
Method tabstop\_set(char\_u \*var, int \*\*array)

```
....  
71.    int n = atoi((char *)cp);
```

### Dangerous Functions\Path 35:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=69">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=69</a>
Status	New

The dangerous function, atoi, was found in use at line 24 in vim@@vim-v8.2.3862-CVE-2022-2183-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c
Line	71	71
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method tabstop\_set(char\_u \*var, int \*\*array)

```
....  
71.    int n = atoi((char *)cp);
```

### Dangerous Functions\Path 36:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=70">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=70</a>
Status	New

The dangerous function, atoi, was found in use at line 2806 in vim@@vim-v8.2.3862-CVE-2022-2264-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c
Line	2835	2835
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2264-TP.c  
Method write\_reg\_contents\_ex(

```
.....
2835.                int    num = atoi((char *)str);
```

### Dangerous Functions\Path 37:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=71">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=71</a>
Status	New

The dangerous function, atoi, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-2287-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	2473	2473
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
.....
2473.                if (atoi((char *)items[1]) == 0)
```

### Dangerous Functions\Path 38:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=72">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=72</a>
Status	New

The dangerous function, atoi, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-2287-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	2503	2503
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
.....  
2503.                compmax = atoi((char *)items[1]);
```

### Dangerous Functions\Path 39:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=73">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=73</a>
Status	New

The dangerous function, atoi, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-2287-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	2511	2511
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
.....  
2511.                compminlen = atoi((char *)items[1]);
```

### Dangerous Functions\Path 40:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=74">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=74</a>
Status	New

The dangerous function, atoi, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-2287-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	2519	2519
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
.....  
2519.                compsyslmax = atoi((char *)items[1]);
```

### Dangerous Functions\Path 41:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=75">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=75</a>
Status	New

The dangerous function, atoi, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-2287-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	2542	2542
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
.....  
2542.                if (atoi((char *)items[1]) == 0)
```

### Dangerous Functions\Path 42:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=76">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=76</a>
Status	New

The dangerous function, atoi, was found in use at line 2186 in vim@@vim-v8.2.3862-CVE-2022-2287-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	2688	2688
Object	atoi	atoi

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
.....
2688.                aff_todo = atoi((char *)items[3]);
```

## 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	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=12">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=12</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 3073 of vim@@vim-v8.2.3862-CVE-2022-1735-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1735-TP.c	vim@@vim-v8.2.3862-CVE-2022-1735-TP.c
Line	3078	3078
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1735-TP.c  
Method replace\_join(

```
.....
3078.                for (i = replace_stack_nr; --i >= 0; )
```

#### Integer Overflow\Path 2:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=13">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=13</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 2886 of vim@@vim-v8.2.3862-CVE-2022-1886-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

Source	Destination
--------	-------------



File	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c
Line	3006	3006
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1886-TP.c  
Method str\_to\_reg(

```
....  
3006.          extra += i;
```

#### Integer Overflow\Path 3:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=14">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=14</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 1511 of vim@@vim-v8.2.3862-CVE-2022-1886-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c
Line	1881	1881
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1886-TP.c  
Method do\_put(

```
....  
1881.          totlen = count * (yanklen + spaces) + bd.startspaces +  
bd.endspaces;
```

#### Integer Overflow\Path 4:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=15">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=15</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 1511 of vim@@vim-v8.2.3862-CVE-2022-1886-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

Source	Destination
--------	-------------

File	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c
Line	2022	2022
Object	AssignExpr	AssignExpr

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-1886-TP.c  
Method do\_put(

```
....  
2022.                totlen = count * yanklen;
```

**Integer Overflow\Path 5:**

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=16">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=16</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 1511 of vim@@vim-v8.2.3862-CVE-2022-1886-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c
Line	1864	1864
Object	AssignExpr	AssignExpr

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-1886-TP.c  
Method do\_put(

```
....  
1864.                spaces = y_width + 1;
```

**Integer Overflow\Path 6:**

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=17">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=17</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 3073 of vim@@vim-v8.2.3862-CVE-2022-2207-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

Source	Destination
--------	-------------

File	vim@@vim-v8.2.3862-CVE-2022-2207-TP.c	vim@@vim-v8.2.3862-CVE-2022-2207-TP.c
Line	3078	3078
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2207-TP.c  
Method replace\_join(

```
....  
3078.         for (i = replace_stack_nr; --i >= 0; )
```

#### Integer Overflow\Path 7:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=18">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=18</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 278 of vim@@vim-v8.2.3862-CVE-2022-2208-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c
Line	309	309
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c  
Method diff\_mark\_adjust\_tp(

```
....  
309.         inserted = amount;
```

#### Integer Overflow\Path 8:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=19">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=19</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 278 of vim@@vim-v8.2.3862-CVE-2022-2208-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

Source	Destination
--------	-------------

File	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c
Line	315	315
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c  
Method diff\_mark\_adjust\_tp(

```
....
315.         inserted = amount_after;
```

#### Integer Overflow\Path 9:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=20">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=20</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 278 of vim@@vim-v8.2.3862-CVE-2022-2208-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c
Line	322	322
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c  
Method diff\_mark\_adjust\_tp(

```
....
322.         deleted = -amount_after;
```

#### Integer Overflow\Path 10:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=21">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=21</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 278 of vim@@vim-v8.2.3862-CVE-2022-2210-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

Source	Destination
--------	-------------

File	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c
Line	309	309
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c  
Method diff\_mark\_adjust\_tp(

```
....  
309.         inserted = amount;
```

#### Integer Overflow\Path 11:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=22">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=22</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 278 of vim@@vim-v8.2.3862-CVE-2022-2210-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c
Line	315	315
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c  
Method diff\_mark\_adjust\_tp(

```
....  
315.         inserted = amount_after;
```

#### Integer Overflow\Path 12:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=23">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=23</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 278 of vim@@vim-v8.2.3862-CVE-2022-2210-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

Source	Destination
--------	-------------

File	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c
Line	322	322
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c  
Method diff\_mark\_adjust\_tp(

```
....  
322.         deleted = -amount_after;
```

#### Integer Overflow\Path 13:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=24">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=24</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 2886 of vim@@vim-v8.2.3862-CVE-2022-2264-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c
Line	3006	3006
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2264-TP.c  
Method str\_to\_reg(

```
....  
3006.         extra += i;
```

#### Integer Overflow\Path 14:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=25">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=25</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 1511 of vim@@vim-v8.2.3862-CVE-2022-2264-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

Source	Destination
--------	-------------

File	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c
Line	1881	1881
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2264-TP.c  
Method do\_put(

```
....
1881.          totlen = count * (yanklen + spaces) + bd.startspaces +
bd.endspace;
```

#### Integer Overflow\Path 15:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=26">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=26</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 1511 of vim@@vim-v8.2.3862-CVE-2022-2264-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c
Line	2022	2022
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2264-TP.c  
Method do\_put(

```
....
2022.          totlen = count * yanklen;
```

#### Integer Overflow\Path 16:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=27">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=27</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 1511 of vim@@vim-v8.2.3862-CVE-2022-2264-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

Source	Destination
--------	-------------

File	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c
Line	1864	1864
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2264-TP.c  
Method do\_put(

```
....  
1864.          spaces = y_width + 1;
```

#### Integer Overflow\Path 17:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=28">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=28</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 278 of vim@@vim-v8.2.3862-CVE-2022-2289-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c
Line	309	309
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2289-TP.c  
Method diff\_mark\_adjust\_tp(

```
....  
309.          inserted = amount;
```

#### Integer Overflow\Path 18:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=29">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=29</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 278 of vim@@vim-v8.2.3862-CVE-2022-2289-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

Source	Destination
--------	-------------



File	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c
Line	315	315
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2289-TP.c  
Method diff\_mark\_adjust\_tp(

```
....  
315.         inserted = amount_after;
```

#### Integer Overflow\Path 19:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=30">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=30</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 278 of vim@@vim-v8.2.3862-CVE-2022-2289-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c
Line	322	322
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2289-TP.c  
Method diff\_mark\_adjust\_tp(

```
....  
322.         deleted = -amount_after;
```

#### Integer Overflow\Path 20:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=31">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=31</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 278 of vim@@vim-v8.2.3862-CVE-2022-2598-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

Source	Destination
--------	-------------

File	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c
Line	309	309
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2598-TP.c  
Method diff\_mark\_adjust\_tp(

```
....  
309.          inserted = amount;
```

#### Integer Overflow\Path 21:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=32">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=32</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 278 of vim@@vim-v8.2.3862-CVE-2022-2598-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c
Line	315	315
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2598-TP.c  
Method diff\_mark\_adjust\_tp(

```
....  
315.          inserted = amount_after;
```

#### Integer Overflow\Path 22:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=33">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=33</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 278 of vim@@vim-v8.2.3862-CVE-2022-2598-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

Source	Destination
--------	-------------

File	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c
Line	322	322
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2598-TP.c  
Method diff\_mark\_adjust\_tp(

```
....
322.         deleted = -amount_after;
```

#### Integer Overflow\Path 23:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=34">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=34</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 3073 of vim@@vim-v8.2.3862-CVE-2022-2845-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2845-TP.c	vim@@vim-v8.2.3862-CVE-2022-2845-TP.c
Line	3078	3078
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2845-TP.c  
Method replace\_join(

```
....
3078.         for (i = replace_stack_nr; --i >= 0; )
```

## Use of Uninitialized Pointer

Query Path:

CPP\Cx\CPP Medium Threat\Use of Uninitialized Pointer Version:0

### Categories

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

### Description

#### Use of Uninitialized Pointer\Path 1:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50</a>

[069&pathid=78](#)

Status New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-1851-TP.c in line 810 is not initialized when it is used by w\_old\_cursor\_lnum at vim@@vim-v8.2.3862-CVE-2022-1851-TP.c in line 810.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c
Line	861	869
Object	wp	w_old_cursor_lnum

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1851-TP.c

Method op\_format(

```
....  
861.          win_T *wp;  
....  
869.          if (wp->w_old_cursor_lnum > wp->w_old_visual_lnum)
```

#### Use of Uninitialized Pointer\Path 2:

Severity Medium

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=79>

Status New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-1851-TP.c in line 810 is not initialized when it is used by w\_old\_cursor\_lnum at vim@@vim-v8.2.3862-CVE-2022-1851-TP.c in line 810.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c
Line	861	865
Object	wp	w_old_cursor_lnum

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1851-TP.c

Method op\_format(

```
....  
861.          win_T *wp;  
....  
865.          if (wp->w_old_cursor_lnum != 0)
```

#### Use of Uninitialized Pointer\Path 3:

Severity Medium

Result State To Verify

Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=80">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=80</a>
Status	New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-1851-TP.c in line 810 is not initialized when it is used by w\_old\_visual\_lnum at vim@@vim-v8.2.3862-CVE-2022-1851-TP.c in line 810.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c
Line	861	869
Object	wp	w_old_visual_lnum

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1851-TP.c  
Method op\_format(

```

.....
861.          win_T *wp;
.....
869.          if (wp->w_old_cursor_lnum > wp->w_old_visual_lnum)

```

#### Use of Uninitialized Pointer\Path 4:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=81">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=81</a>
Status	New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-1851-TP.c in line 810 is not initialized when it is used by w\_old\_cursor\_lnum at vim@@vim-v8.2.3862-CVE-2022-1851-TP.c in line 810.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c
Line	861	870
Object	wp	w_old_cursor_lnum

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1851-TP.c  
Method op\_format(

```

.....
861.          win_T *wp;
.....
870.          wp->w_old_cursor_lnum += old_line_count;

```

#### Use of Uninitialized Pointer\Path 5:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=82">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=82</a>
Status	New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-1851-TP.c in line 810 is not initialized when it is used by w\_old\_visual\_lnum at vim@@vim-v8.2.3862-CVE-2022-1851-TP.c in line 810.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c
Line	861	872
Object	wp	w_old_visual_lnum

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1851-TP.c  
Method op\_format(

```
....  
861.          win_T *wp;  
....  
872.          wp->w_old_visual_lnum += old_line_count;
```

### Use of Uninitialized Pointer\Path 6:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=83">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=83</a>
Status	New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-2208-TP.c in line 124 is not initialized when it is used by w\_p\_diff at vim@@vim-v8.2.3862-CVE-2022-2208-TP.c in line 124.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c
Line	126	134
Object	wp	w_p_diff

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c  
Method diff\_buf\_adjust(win\_T \*win)

```
....  
126.          win_T *wp;  
....  
134.          if (wp->w_buffer == win->w_buffer && wp->w_p_diff)
```

**Use of Uninitialized Pointer\Path 7:**

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=84">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=84</a>
Status	New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-2208-TP.c in line 124 is not initialized when it is used by w\_buffer at vim@@vim-v8.2.3862-CVE-2022-2208-TP.c in line 124.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c
Line	126	134
Object	wp	w_buffer

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c  
Method diff\_buf\_adjust(win\_T \*win)

```
....  
126.      win_T      *wp;  
....  
134.      if (wp->w_buffer == win->w_buffer && wp->w_p_diff)
```

**Use of Uninitialized Pointer\Path 8:**

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=85">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=85</a>
Status	New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-2208-TP.c in line 124 is not initialized when it is used by wp at vim@@vim-v8.2.3862-CVE-2022-2208-TP.c in line 124.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c
Line	126	136
Object	wp	wp

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c  
Method diff\_buf\_adjust(win\_T \*win)

```
....
126.      win_T      *wp;
....
136.      if (wp == NULL)
```

### Use of Uninitialized Pointer\Path 9:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=86">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=86</a>
Status	New

The variable declared in tp at vim@@vim-v8.2.3862-CVE-2022-2208-TP.c in line 2207 is not initialized when it is used by tp\_diff\_invalid at vim@@vim-v8.2.3862-CVE-2022-2208-TP.c in line 2207.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c
Line	2215	2338
Object	tp	tp_diff_invalid

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c  
Method diffopt\_changed(void)

```
....
2215.      tabpage_T      *tp;
....
2338.      tp->tp_diff_invalid = TRUE;
```

### Use of Uninitialized Pointer\Path 10:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=87">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=87</a>
Status	New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-2210-TP.c in line 124 is not initialized when it is used by w\_p\_diff at vim@@vim-v8.2.3862-CVE-2022-2210-TP.c in line 124.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c
Line	126	134
Object	wp	w_p_diff

#### Code Snippet



File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c  
Method diff\_buf\_adjust(win\_T \*win)

```
....
126.      win_T      *wp;
....
134.      if (wp->w_buffer == win->w_buffer && wp->w_p_diff)
```

#### Use of Uninitialized Pointer\Path 11:

Severity Medium  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=88>  
Status New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-2210-TP.c in line 124 is not initialized when it is used by w\_buffer at vim@@vim-v8.2.3862-CVE-2022-2210-TP.c in line 124.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c
Line	126	134
Object	wp	w_buffer

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c  
Method diff\_buf\_adjust(win\_T \*win)

```
....
126.      win_T      *wp;
....
134.      if (wp->w_buffer == win->w_buffer && wp->w_p_diff)
```

#### Use of Uninitialized Pointer\Path 12:

Severity Medium  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=89>  
Status New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-2210-TP.c in line 124 is not initialized when it is used by wp at vim@@vim-v8.2.3862-CVE-2022-2210-TP.c in line 124.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c
Line	126	136
Object	wp	wp

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c  
Method diff\_buf\_adjust(win\_T \*win)

```
....
126.      win_T      *wp;
....
136.      if (wp == NULL)
```

#### Use of Uninitialized Pointer\Path 13:

Severity Medium  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=90>  
Status New

The variable declared in tp at vim@@vim-v8.2.3862-CVE-2022-2210-TP.c in line 2207 is not initialized when it is used by tp\_diff\_invalid at vim@@vim-v8.2.3862-CVE-2022-2210-TP.c in line 2207.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c
Line	2215	2338
Object	tp	tp_diff_invalid

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c  
Method diffopt\_changed(void)

```
....
2215.      tabpage_T      *tp;
....
2338.      tp->tp_diff_invalid = TRUE;
```

#### Use of Uninitialized Pointer\Path 14:

Severity Medium  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=91>  
Status New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-2289-TP.c in line 124 is not initialized when it is used by w\_p\_diff at vim@@vim-v8.2.3862-CVE-2022-2289-TP.c in line 124.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c

Line	126	134
Object	wp	w_p_diff

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2289-TP.c

Method diff\_buf\_adjust(win\_T \*win)

```
....
126.      win_T      *wp;
....
134.      if (wp->w_buffer == win->w_buffer && wp->w_p_diff)
```

#### Use of Uninitialized Pointer\Path 15:

Severity Medium

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=92>

Status New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-2289-TP.c in line 124 is not initialized when it is used by w\_buffer at vim@@vim-v8.2.3862-CVE-2022-2289-TP.c in line 124.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c
Line	126	134
Object	wp	w_buffer

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2289-TP.c

Method diff\_buf\_adjust(win\_T \*win)

```
....
126.      win_T      *wp;
....
134.      if (wp->w_buffer == win->w_buffer && wp->w_p_diff)
```

#### Use of Uninitialized Pointer\Path 16:

Severity Medium

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=93>

Status New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-2289-TP.c in line 124 is not initialized when it is used by wp at vim@@vim-v8.2.3862-CVE-2022-2289-TP.c in line 124.

Source	Destination
--------	-------------

File	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c
Line	126	136
Object	wp	wp

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2289-TP.c  
Method diff\_buf\_adjust(win\_T \*win)

```
....
126.      win_T      *wp;
....
136.      if (wp == NULL)
```

#### Use of Uninitialized Pointer\Path 17:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=94">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=94</a>
Status	New

The variable declared in tp at vim@@vim-v8.2.3862-CVE-2022-2289-TP.c in line 2207 is not initialized when it is used by tp\_diff\_invalid at vim@@vim-v8.2.3862-CVE-2022-2289-TP.c in line 2207.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c
Line	2215	2338
Object	tp	tp_diff_invalid

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2289-TP.c  
Method diffopt\_changed(void)

```
....
2215.      tabpage_T      *tp;
....
2338.      tp->tp_diff_invalid = TRUE;
```

#### Use of Uninitialized Pointer\Path 18:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=95">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=95</a>
Status	New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-2598-TP.c in line 124 is not initialized when it is used by w\_p\_diff at vim@@vim-v8.2.3862-CVE-2022-2598-TP.c in line 124.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c
Line	126	134
Object	wp	w_p_diff

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2598-TP.c  
Method diff\_buf\_adjust(win\_T \*win)

```
....
126.      win_T      *wp;
....
134.          if (wp->w_buffer == win->w_buffer && wp->w_p_diff)
```

#### Use of Uninitialized Pointer\Path 19:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=96">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=96</a>
Status	New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-2598-TP.c in line 124 is not initialized when it is used by w\_buffer at vim@@vim-v8.2.3862-CVE-2022-2598-TP.c in line 124.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c
Line	126	134
Object	wp	w_buffer

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2598-TP.c  
Method diff\_buf\_adjust(win\_T \*win)

```
....
126.      win_T      *wp;
....
134.          if (wp->w_buffer == win->w_buffer && wp->w_p_diff)
```

#### Use of Uninitialized Pointer\Path 20:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=97">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=97</a>
Status	New

The variable declared in wp at vim@@vim-v8.2.3862-CVE-2022-2598-TP.c in line 124 is not initialized when it is used by wp at vim@@vim-v8.2.3862-CVE-2022-2598-TP.c in line 124.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c
Line	126	136
Object	wp	wp

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2598-TP.c  
Method diff\_buf\_adjust(win\_T \*win)

```
....
126.      win_T      *wp;
....
136.      if (wp == NULL)
```

#### Use of Uninitialized Pointer\Path 21:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=98">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=98</a>
Status	New

The variable declared in tp at vim@@vim-v8.2.3862-CVE-2022-2598-TP.c in line 2207 is not initialized when it is used by tp at vim@@vim-v8.2.3862-CVE-2022-2598-TP.c in line 2207.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c
Line	2215	2338
Object	tp	tp

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2598-TP.c  
Method diffopt\_changed(void)

```
....
2215.      tabpage_T      *tp;
....
2338.      tp->tp_diff_invalid = TRUE;
```

## Divide By Zero

Query Path:

CPP\Cx\CPP Medium Threat\Divide By Zero Version:1

[Description](#)

#### Divide By Zero\Path 1:

Severity	Medium
----------	--------

Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=1">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=1</a>
Status	New

The application performs an illegal operation in num\_divide, in vim@@vim-v8.2.3862-CVE-2022-1420-FP.c. In line 66, the program attempts to divide by n2, which might be evaluate to 0 (zero) at time of division. This value could be a hard-coded zero value, or received from external, untrusted input n2 in num\_divide of vim@@vim-v8.2.3862-CVE-2022-1420-FP.c, at line 66.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	86	86
Object	n2	n2

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method num\_divide(varnumber\_T n1, varnumber\_T n2, int \*failed)

```
....  
86.    result = n1 / n2;
```

#### Divide By Zero\Path 2:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=2">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=2</a>
Status	New

The application performs an illegal operation in parse\_cino, in vim@@vim-v8.2.3862-CVE-2022-1733-TP.c. In line 1709, the program attempts to divide by divider, which might be evaluate to 0 (zero) at time of division. This value could be a hard-coded zero value, or received from external, untrusted input divider in parse\_cino of vim@@vim-v8.2.3862-CVE-2022-1733-TP.c, at line 1709.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1733-TP.c	vim@@vim-v8.2.3862-CVE-2022-1733-TP.c
Line	1879	1879
Object	divider	divider

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1733-TP.c  
Method parse\_cino(buf\_T \*buf)

```
....  
1879.                n += (sw * fraction + divider / 2) / divider;
```

**Divide By Zero\Path 3:**

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=3">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=3</a>
Status	New

The application performs an illegal operation in `tabstop_padding`, in `vim@@vim-v8.2.3862-CVE-2022-2125-TP.c`. In line 97, the program attempts to divide by `ts`, which might be evaluate to 0 (zero) at time of division. This value could be a hard-coded zero value, or received from external, untrusted input `ts` in `tabstop_padding` of `vim@@vim-v8.2.3862-CVE-2022-2125-TP.c`, at line 97.

	Source	Destination
File	<code>vim@@vim-v8.2.3862-CVE-2022-2125-TP.c</code>	<code>vim@@vim-v8.2.3862-CVE-2022-2125-TP.c</code>
Line	106	106
Object	<code>ts</code>	<code>ts</code>

**Code Snippet**

File Name `vim@@vim-v8.2.3862-CVE-2022-2125-TP.c`  
Method `tabstop_padding(colnr_T col, int ts_arg, int *vts)`

```
....  
106.         return ts - (col % ts);
```

**Divide By Zero\Path 4:**

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=4">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=4</a>
Status	New

The application performs an illegal operation in `tabstop_padding`, in `vim@@vim-v8.2.3862-CVE-2022-2183-TP.c`. In line 97, the program attempts to divide by `ts`, which might be evaluate to 0 (zero) at time of division. This value could be a hard-coded zero value, or received from external, untrusted input `ts` in `tabstop_padding` of `vim@@vim-v8.2.3862-CVE-2022-2183-TP.c`, at line 97.

	Source	Destination
File	<code>vim@@vim-v8.2.3862-CVE-2022-2183-TP.c</code>	<code>vim@@vim-v8.2.3862-CVE-2022-2183-TP.c</code>
Line	106	106
Object	<code>ts</code>	<code>ts</code>

**Code Snippet**

File Name `vim@@vim-v8.2.3862-CVE-2022-2183-TP.c`  
Method `tabstop_padding(colnr_T col, int ts_arg, int *vts)`



```
....
106.         return ts - (col % ts);
```

### Divide By Zero\Path 5:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=5">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=5</a>
Status	New

The application performs an illegal operation in num\_divide, in vim@@vim-v8.2.3862-CVE-2022-2231-FP.c. In line 66, the program attempts to divide by n2, which might be evaluate to 0 (zero) at time of division. This value could be a hard-coded zero value, or received from external, untrusted input n2 in num\_divide of vim@@vim-v8.2.3862-CVE-2022-2231-FP.c, at line 66.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	86	86
Object	n2	n2

### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method num\_divide(varnumber\_T n1, varnumber\_T n2, int \*failed)

```
....
86.     result = n1 / n2;
```

## 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	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=7">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=7</a>
Status	New

Calling free() (line 4681) on a variable that was not dynamically allocated (line 4681) in file vim@@vim-v8.2.3862-CVE-2022-1420-FP.c may result with a crash.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4715	4715

Object	tempitem	tempitem
--------	----------	----------

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_ht(hashtab\_T \*ht, int copyID, list\_stack\_T \*\*list\_stack)

```
....  
4715.      free(tempitem);
```

#### MemoryFree on StackVariable\Path 2:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=8">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=8</a>
Status	New

Calling free() (line 4758) on a variable that was not dynamically allocated (line 4758) in file vim@@vim-v8.2.3862-CVE-2022-1420-FP.c may result with a crash.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4783	4783
Object	tempitem	tempitem

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_list\_items(list\_T \*l, int copyID, ht\_stack\_T \*\*ht\_stack)

```
....  
4783.      free(tempitem);
```

#### MemoryFree on StackVariable\Path 3:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=9">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=9</a>
Status	New

Calling free() (line 4681) on a variable that was not dynamically allocated (line 4681) in file vim@@vim-v8.2.3862-CVE-2022-2231-FP.c may result with a crash.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	4715	4715

Object	tempitem	tempitem
--------	----------	----------

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method set\_ref\_in\_ht(hashtab\_T \*ht, int copyID, list\_stack\_T \*\*list\_stack)

```
....  
4715.         free(tempitem);
```

#### MemoryFree on StackVariable\Path 4:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=10">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=10</a>
Status	New

Calling free() (line 4758) on a variable that was not dynamically allocated (line 4758) in file vim@@vim-v8.2.3862-CVE-2022-2231-FP.c may result with a crash.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	4783	4783
Object	tempitem	tempitem

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method set\_ref\_in\_list\_items(list\_T \*l, int copyID, ht\_stack\_T \*\*ht\_stack)

```
....  
4783.         free(tempitem);
```

## Use of Uninitialized Variable

Query Path:

CPP\Cx\CPP Medium Threat\Use of Uninitialized Variable Version:0

### Categories

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

### Description

#### Use of Uninitialized Variable\Path 1:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=99">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=99</a>
Status	New

Source	Destination
--------	-------------

File	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c
Line	1437	1566
Object	ind_len	ind_len

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c  
Method copy\_indent(int size, char\_u \*src)

```
....
1437.      int          ind_len;
....
1566.      curwin->w_cursor.col = ind_len;
```

#### Use of Uninitialized Variable\Path 2:

Severity	Medium
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=100">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=100</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c
Line	1437	1566
Object	ind_len	ind_len

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method copy\_indent(int size, char\_u \*src)

```
....
1437.      int          ind_len;
....
1566.      curwin->w_cursor.col = ind_len;
```

## 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	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=6">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=6</a>
Status	New

The pointer trypos at vim@@vim-v8.2.3862-CVE-2022-1733-TP.c in line 1638 is being used after it has been freed.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1733-TP.c	vim@@vim-v8.2.3862-CVE-2022-1733-TP.c
Line	1660	1660
Object	trypos	trypos

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1733-TP.c  
Method find\_start\_brace(void) // XXX

```
....
1660.     return trypos;
```

## 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	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=11">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=11</a>
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 652 of vim@@vim-v8.2.3862-CVE-2022-2889-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	691	691
Object	AssignExpr	AssignExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c  
Method get\_function\_body(

```
....
691.      nesting_def[0] = vim9_function;
```

## 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	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=77">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=77</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c	vim@@vim-v8.2.3862-CVE-2022-1851-TP.c
Line	634	634
Object	neW	neW

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1851-TP.c  
Method auto\_format(

```
....
634.      char_u  *new, *pnew;
```

## 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	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=386">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=386</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 247 is not initialized when it is used by skip at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 204.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	299	212
Object	null	skip

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method eval\_expr\_typval(typval\_T \*expr, typval\_T \*argv, int argc, typval\_T \*rettv)

```
....  
299.          if (eval1_emsg(&s, rettv, NULL) == FAIL)
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method eval1\_emsg(char\_u \*\*arg, typval\_T \*rettv, exarg\_T \*eap)

```
....  
212.          fill_evalarg_from_eap(&evalarg, eap, eap != NULL && eap->skip);
```

#### NULL Pointer Dereference\Path 2:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=387>  
Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 386 is not initialized when it is used by Pointer at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 386.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	443	443
Object	null	Pointer

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method skip\_expr\_concatenate(

```
....  
443.          *((char_u **)gap->ga_data) = NULL;
```

**NULL Pointer Dereference\Path 3:**

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=388">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=388</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 544 is not initialized when it is used by skip at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 521.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	548	530
Object	null	skip

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method eval\_to\_string(

```
....  
548.      return eval_to_string_eap(arg, convert, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method eval\_to\_string\_eap(

```
....  
530.      fill_evalarg_from_eap(&evalarg, eap, eap != NULL && eap->skip);
```

**NULL Pointer Dereference\Path 4:**

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=389">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=389</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by ht\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4506	4842
Object	null	ht_stack



#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method garbage\_collect(int testing)

```
....
4506.                                     NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method set\_ref\_in\_item(

```
....
4842.                                     newitem->prev = *ht_stack;
```

#### NULL Pointer Dereference\Path 5:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=390>

Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by ht\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4511	4842
Object	null	ht_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method garbage\_collect(int testing)

```
....
4511.                                     NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method set\_ref\_in\_item(

```
....
4842.                                     newitem->prev = *ht_stack;
```

#### NULL Pointer Dereference\Path 6:

Severity Low

Result State To Verify

Online Results <http://WIN->

[PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=391](http://PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=391)

Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by ht\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4514	4842
Object	null	ht_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method garbage\_collect(int testing)

```
....  
4514.                                     NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```
....  
4842.                                     newitem->prev = *ht_stack;
```

#### NULL Pointer Dereference\Path 7:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=392>  
Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by ht\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4518	4842
Object	null	ht_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method garbage\_collect(int testing)

```
.....
4518.                                     NULL, NULL) ;
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```
.....
4842.                                     newitem->prev = *ht_stack;
```

### NULL Pointer Dereference\Path 8:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=393>  
Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by ht\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4522	4842
Object	null	ht_stack

### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method garbage\_collect(int testing)

```
.....
4522.                                     NULL, NULL) ;
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```
.....
4842.                                     newitem->prev = *ht_stack;
```

### NULL Pointer Dereference\Path 9:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=394>  
Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by ht\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4528	4842
Object	null	ht_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method garbage\_collect(int testing)

```
....
4528.                                     NULL, NULL) ;
```



File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method set\_ref\_in\_item(

```
....
4842.                                newitem->prev = *ht_stack;
```

#### NULL Pointer Dereference\Path 10:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=395>

Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4793 is not initialized when it is used by ht\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4802	4842
Object	null	ht_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method set\_ref\_in\_callback(callback\_T \*cb, int copyID)

```
....
4802.        return set_ref_in_item(&tv, copyID, NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```
....  
4842. newitem->prev = *ht_stack;
```

### NULL Pointer Dereference\Path 11:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=396>  
Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by prev at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4528	4842
Object	null	prev

### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method garbage\_collect(int testing)

```
....  
4528. NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```
....  
4842. newitem->prev = *ht_stack;
```

### NULL Pointer Dereference\Path 12:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=397>  
Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by prev at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4514	4842
Object	null	prev

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method garbage\_collect(int testing)

```
....
4514.                                     NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```
....
4842.                                newitem->prev = *ht_stack;
```

### NULL Pointer Dereference\Path 13:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=398">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=398</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by prev at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4522	4842
Object	null	prev

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method garbage\_collect(int testing)

```
....
4522.                                     NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```
.....
4842.                                newitem->prev = *ht_stack;
```

#### NULL Pointer Dereference\Path 14:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=399">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=399</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by prev at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4518	4842
Object	null	prev

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method garbage\_collect(int testing)

```
.....
4518.                                NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```
.....
4842.                                newitem->prev = *ht_stack;
```

#### NULL Pointer Dereference\Path 15:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=400">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=400</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by prev at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Line	4511	4842
Object	null	prev

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method garbage\_collect(int testing)

```
....
4511.                                     NULL, NULL);
```



File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method set\_ref\_in\_item(

```
....
4842.                                newitem->prev = *ht_stack;
```

#### NULL Pointer Dereference\Path 16:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=401>

Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4793 is not initialized when it is used by prev at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4802	4842
Object	null	prev

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method set\_ref\_in\_callback(callback\_T \*cb, int copyID)

```
....
4802.        return set_ref_in_item(&tv, copyID, NULL, NULL);
```



File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method set\_ref\_in\_item(

```
....
4842.                                newitem->prev = *ht_stack;
```



### NULL Pointer Dereference\Path 17:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=402">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=402</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by prev at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4506	4842
Object	null	prev

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method garbage\_collect(int testing)

```
....
4506.                                     NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```
....
4842.                                newitem->prev = *ht_stack;
```

### NULL Pointer Dereference\Path 18:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=403">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=403</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by list\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4506	4869
Object	null	list_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method garbage\_collect(int testing)

```
....
4506.                                     NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method set\_ref\_in\_item(

```
....
4869.                                     newitem->prev = *list_stack;
```

#### NULL Pointer Dereference\Path 19:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=404>

Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by list\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4511	4869
Object	null	list_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method garbage\_collect(int testing)

```
....
4511.                                     NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method set\_ref\_in\_item(

```
....
4869.                                     newitem->prev = *list_stack;
```

#### NULL Pointer Dereference\Path 20:

Severity Low

Result State To Verify

Online Results <http://WIN->

[PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=405](http://PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=405)

Status New


The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by list\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4514	4869
Object	null	list_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method garbage\_collect(int testing)

```
....  
4514.                                     NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(  


```
....  
4869.                                newitem->prev = *list_stack;
```

#### NULL Pointer Dereference\Path 21:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=406>  
Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by list\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4518	4869
Object	null	list_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method garbage\_collect(int testing)

```

.....
4518.                                     NULL, NULL) ;

```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```

.....
4869.                                     newitem->prev = *list_stack;

```

### NULL Pointer Dereference\Path 22:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=407>  
Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by list\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4522	4869
Object	null	list_stack

### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method garbage\_collect(int testing)

```

.....
4522.                                     NULL, NULL) ;

```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```

.....
4869.                                     newitem->prev = *list_stack;

```

### NULL Pointer Dereference\Path 23:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=408>  
Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by list\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4528	4869
Object	null	list_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method garbage\_collect(int testing)

```
....
4528.                                     NULL, NULL) ;
```



File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method set\_ref\_in\_item(

```
....
4869.                                     newitem->prev = *list_stack;
```

#### NULL Pointer Dereference\Path 24:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=409>

Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4793 is not initialized when it is used by list\_stack at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4802	4869
Object	null	list_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method set\_ref\_in\_callback(callback\_T \*cb, int copyID)

```
....
4802.         return set_ref_in_item(&tv, copyID, NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```
....  
4869. newitem->prev = *list_stack;
```

### NULL Pointer Dereference\Path 25:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=410>  
Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by prev at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4514	4869
Object	null	prev

### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method garbage\_collect(int testing)

```
....  
4514. NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```
....  
4869. newitem->prev = *list_stack;
```

### NULL Pointer Dereference\Path 26:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=411>  
Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4793 is not initialized when it is used by prev at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4802	4869
Object	null	prev

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_callback(callback\_T \*cb, int copyID)

```
....
4802.         return set_ref_in_item(&tv, copyID, NULL, NULL);
```



File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```
....
4869.                                newitem->prev = *list_stack;
```

#### NULL Pointer Dereference\Path 27:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=412">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=412</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by prev at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4528	4869
Object	null	prev

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method garbage\_collect(int testing)

```
....
4528.                                NULL, NULL);
```



File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```
.....
4869.                                newitem->prev = *list_stack;
```

### NULL Pointer Dereference\Path 28:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=413">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=413</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by prev at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4522	4869
Object	null	prev

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method garbage\_collect(int testing)

```
.....
4522.                                NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(

```
.....
4869.                                newitem->prev = *list_stack;
```

### NULL Pointer Dereference\Path 29:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=414">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=414</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by prev at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c



Line	4511	4869
Object	null	prev

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method garbage\_collect(int testing)

```
....
4511.                                     NULL, NULL);
```



File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method set\_ref\_in\_item(

```
....
4869.                                newitem->prev = *list_stack;
```

#### NULL Pointer Dereference\Path 30:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=415>

Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by prev at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4506	4869
Object	null	prev

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method garbage\_collect(int testing)

```
....
4506.                                     NULL, NULL);
```



File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c

Method set\_ref\_in\_item(

```
....
4869.                                newitem->prev = *list_stack;
```

**NULL Pointer Dereference\Path 31:**

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=416">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=416</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4445 is not initialized when it is used by prev at vim@@vim-v8.2.3862-CVE-2022-1420-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	4518	4869
Object	null	prev

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method garbage\_collect(int testing)

```
....  
4518.                                     NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method set\_ref\_in\_item(  
↓

```
....  
4869.                               newitem->prev = *list_stack;
```

**NULL Pointer Dereference\Path 32:**

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=417">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=417</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1621-TP.c in line 5459 is not initialized when it is used by sl\_map\_hash at vim@@vim-v8.2.3862-CVE-2022-1621-TP.c in line 6610.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	5479	6665
Object	null	sl_map_hash

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method spell\_make\_sugfile(spellinfo\_T \*spin, char\_u \*wfname)

```
....
5479.          slang = spell_load_file(wfname, NULL, NULL, FALSE);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method set\_map\_str(slang\_T \*lp, char\_u \*map)

```
....
6665.          hash_add_item(&lp->sl_map_hash, hi, b, hash);
```

### NULL Pointer Dereference\Path 33:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=418">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=418</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1621-TP.c in line 5459 is not initialized when it is used by sl\_map\_hash at vim@@vim-v8.2.3862-CVE-2022-1621-TP.c in line 6610.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	5479	6627
Object	null	sl_map_hash

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method spell\_make\_sugfile(spellinfo\_T \*spin, char\_u \*wfname)

```
....
5479.          slang = spell_load_file(wfname, NULL, NULL, FALSE);
```

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method set\_map\_str(slang\_T \*lp, char\_u \*map)

```
....
6627.          hash_init(&lp->sl_map_hash);
```

### NULL Pointer Dereference\Path 34:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-">http://WIN-</a>

	<a href="http://PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=419">PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=419</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1733-TP.c in line 2034 is not initialized when it is used by col at vim@@vim-v8.2.3862-CVE-2022-1733-TP.c in line 2034.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1733-TP.c	vim@@vim-v8.2.3862-CVE-2022-1733-TP.c
Line	2344	2497
Object	null	col

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1733-TP.c  
Method get\_c\_indent(void)

```
....  
2344.                trypos = NULL;  
....  
2497.                our_paren_pos.col++;
```

#### NULL Pointer Dereference\Path 35:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=420">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=420</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1733-TP.c in line 2034 is not initialized when it is used by col at vim@@vim-v8.2.3862-CVE-2022-1733-TP.c in line 2034.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1733-TP.c	vim@@vim-v8.2.3862-CVE-2022-1733-TP.c
Line	2344	2506
Object	null	col

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1733-TP.c  
Method get\_c\_indent(void)

```
....  
2344.                trypos = NULL;  
....  
2506.                our_paren_pos.col++;
```

#### NULL Pointer Dereference\Path 36:

Severity	Low
----------	-----

Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=421">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=421</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1733-TP.c in line 2034 is not initialized when it is used by sp at vim@@vim-v8.2.3862-CVE-2022-1733-TP.c in line 1159.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1733-TP.c	vim@@vim-v8.2.3862-CVE-2022-1733-TP.c
Line	3666	1174
Object	null	sp

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1733-TP.c  
Method get\_c\_indent(void)

```
....
3666.          && cin_isfuncdecl(NULL, cur_curpos.lnum + 1,
```



File Name vim@@vim-v8.2.3862-CVE-2022-1733-TP.c  
Method cin\_isfuncdecl(

```
....
1174.          s = *sp;
```

#### NULL Pointer Dereference\Path 37:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=422">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=422</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-1733-TP.c in line 2034 is not initialized when it is used by sp at vim@@vim-v8.2.3862-CVE-2022-1733-TP.c in line 1159.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1733-TP.c	vim@@vim-v8.2.3862-CVE-2022-1733-TP.c
Line	3760	1174
Object	null	sp

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1733-TP.c  
Method get\_c\_indent(void)

```
.....
3760.          if (cin_isfuncdecl(NULL, cur_curpos.lnum, 0)) // XXX
```

File Name vim@@vim-v8.2.3862-CVE-2022-1733-TP.c

Method cin\_isfuncdecl(

```
.....
1174.          s = *sp;
```

### NULL Pointer Dereference\Path 38:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=423>

Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 24 is not initialized when it is used by Pointer at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 24.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c
Line	78	77
Object	null	Pointer

### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c

Method tabstop\_set(char\_u \*var, int \*\*array)

```
.....
78.          *array = NULL;
.....
77.          vim_free(*array);
```

### NULL Pointer Dereference\Path 39:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=424>

Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 1574 is not initialized when it is used by Pointer at vim@@vim-v8.2.3862-CVE-2022-2125-TP.c in line 24.

Source	Destination
--------	-------------

File	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c
Line	1590	77
Object	null	Pointer

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c  
Method ex\_retab(exarg\_T \*eap)

```
....
1590.      int      *new_vts_array = NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c  
Method tabstop\_set(char\_u \*var, int \*\*array)

```
....
77.      vim_free(*array);
```

#### NULL Pointer Dereference\Path 40:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=425">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=425</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 24 is not initialized when it is used by Pointer at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 24.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c
Line	78	77
Object	null	Pointer

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method tabstop\_set(char\_u \*var, int \*\*array)

```
....
78.      *array = NULL;
....
77.      vim_free(*array);
```

#### NULL Pointer Dereference\Path 41:

Severity	Low
Result State	To Verify

Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=426">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=426</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 1574 is not initialized when it is used by Pointer at vim@@vim-v8.2.3862-CVE-2022-2183-TP.c in line 24.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c
Line	1590	77
Object	null	Pointer

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method ex\_retab(exarg\_T \*eap)

```
....
1590.      int      *new_vts_array = NULL;
```



File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method tabstop\_set(char\_u \*var, int \*\*array)

```
....
77.      vim_free(*array);
```

#### NULL Pointer Dereference\Path 42:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=427">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=427</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-2208-TP.c in line 654 is not initialized when it is used by w\_buffer at vim@@vim-v8.2.3862-CVE-2022-2208-TP.c in line 2087.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c
Line	658	2183
Object	null	w_buffer

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c  
Method diff\_redraw(



```
....
658.      win_T      *wp_other = NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c  
Method diff\_set\_topline(win\_T \*fromwin, win\_T \*towin)

```
....
2183.      towin->w_topline = towin->w_buffer->b_ml.ml_line_count;
```

### NULL Pointer Dereference\Path 43:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=428>  
Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-2208-TP.c in line 654 is not initialized when it is used by w\_buffer at vim@@vim-v8.2.3862-CVE-2022-2208-TP.c in line 2087.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c
Line	658	2181
Object	null	w_buffer

### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c  
Method diff\_redraw(

```
....
658.      win_T      *wp_other = NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c  
Method diff\_set\_topline(win\_T \*fromwin, win\_T \*towin)

```
....
2181.      if (towin->w_topline > towin->w_buffer->b_ml.ml_line_count)
```

### NULL Pointer Dereference\Path 44:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=429>  
Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-2210-TP.c in line 654 is not initialized when it is used by w\_buffer at vim@@vim-v8.2.3862-CVE-2022-2210-TP.c in line 2087.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c
Line	658	2183
Object	null	w_buffer

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c

Method diff\_redraw(

```
....
658.      win_T      *wp_other = NULL;
```



File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c

Method diff\_set\_topline(win\_T \*fromwin, win\_T \*twin)

```
....
2183.      twin->w_topline = twin->w_buffer->b_ml.ml_line_count;
```

#### NULL Pointer Dereference\Path 45:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=430>

Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-2210-TP.c in line 654 is not initialized when it is used by w\_buffer at vim@@vim-v8.2.3862-CVE-2022-2210-TP.c in line 2087.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c
Line	658	2181
Object	null	w_buffer

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c

Method diff\_redraw(

```
....
658.      win_T      *wp_other = NULL;
```

File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c  
Method diff\_set\_topline(win\_T \*fromwin, win\_T \*twin)

```
....
2181.         if (twin->w_topline > twin->w_buffer->b_ml.ml_line_count)
```

#### NULL Pointer Dereference\Path 46:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=431>  
Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 247 is not initialized when it is used by skip at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 204.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	299	212
Object	null	skip

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method eval\_expr\_typval(typval\_T \*expr, typval\_T \*argv, int argc, typval\_T \*rettv)

```
....
299.         if (eval1_emsg(&s, rettv, NULL) == FAIL)
```

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method eval1\_emsg(char\_u \*\*arg, typval\_T \*rettv, exarg\_T \*eap)

```
....
212.         fill_evalarg_from_eap(&evalarg, eap, eap != NULL && eap-
>skip);
```

#### NULL Pointer Dereference\Path 47:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=432>  
Status New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 386 is not initialized when it is used by Pointer at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 386.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	443	443
Object	null	Pointer

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method skip\_expr\_concatenate(

```
....  
443.          * ((char_u **) gap->ga_data) = NULL;
```

#### NULL Pointer Dereference\Path 48:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=433">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=433</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 544 is not initialized when it is used by skip at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 521.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	548	530
Object	null	skip

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method eval\_to\_string(

```
....  
548.          return eval_to_string_eap(arg, convert, NULL);
```



File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method eval\_to\_string\_eap(

```
....  
530.          fill_evalarg_from_eap(&evalarg, eap, eap != NULL && eap->skip);
```

#### NULL Pointer Dereference\Path 49:

Severity	Low
Result State	To Verify

Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=434">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=434</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 4445 is not initialized when it is used by ht\_stack at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	4506	4842
Object	null	ht_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method garbage\_collect(int testing)

```
....  
4506.                                     NULL, NULL);
```



File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method set\_ref\_in\_item(

```
....  
4842.                               newitem->prev = *ht_stack;
```

#### NULL Pointer Dereference\Path 50:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=435">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=435</a>
Status	New

The variable declared in null at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 4445 is not initialized when it is used by ht\_stack at vim@@vim-v8.2.3862-CVE-2022-2231-FP.c in line 4813.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	4511	4842
Object	null	ht_stack

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method garbage\_collect(int testing)

```
.....
4511.                                     NULL, NULL);
```

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c

Method set\_ref\_in\_item(

```
.....
4842.                                     newitem->prev = *ht_stack;
```

## 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	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=608">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=608</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	6323	6323
Object	fprintf	fprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c

Method spell\_add\_word(

```
.....
6323.                                     fprintf(fd, "%.*s/!\n", len, word);
```

#### Improper Resource Access Authorization\Path 2:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=609">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=609</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	6325	6325
Object	fprintf	fprintf

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method spell\_add\_word(

```
....  
6325.                fprintf(fd, "%.*s/?\n", len, word);
```

**Improper Resource Access Authorization\Path 3:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=610>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	6327	6327
Object	fprintf	fprintf

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method spell\_add\_word(

```
....  
6327.                fprintf(fd, "%.*s\n", len, word);
```

**Improper Resource Access Authorization\Path 4:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=611>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	6323	6323

Object	fprintf	fprintf
--------	---------	---------

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c

Method spell\_add\_word(

```
....  
6323.                fprintf(fd, "%.*s/!\n", len, word);
```

#### Improper Resource Access Authorization\Path 5:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=612>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	6325	6325
Object	fprintf	fprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c

Method spell\_add\_word(

```
....  
6325.                fprintf(fd, "%.*s/?\n", len, word);
```

#### Improper Resource Access Authorization\Path 6:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=613>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	6327	6327
Object	fprintf	fprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c

Method spell\_add\_word(



```
....  
6327.                fprintf(fd, "%.s\n", len, word);
```

### Improper Resource Access Authorization\Path 7:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=614">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=614</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c
Line	2672	2672
Object	fprintf	fprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2581-TP.c  
Method vim\_regcomp(char\_u \*expr\_arg, int re\_flags)

```
....  
2672.                fprintf(f, "Syntax error in \"%s\"\n", expr);
```

### Improper Resource Access Authorization\Path 8:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=615">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=615</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	4947	4947
Object	fputc	fputc

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
4947.                fputc(128, fd);                                // <charflagslen>
```

### Improper Resource Access Authorization\Path 9:

Severity	Low
----------	-----

Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=616">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=616</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	4955	4955
Object	fputc	fputc

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
4955.                fputc(flags, fd);                // <charflags>
```

### Improper Resource Access Authorization\Path 10:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=617">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=617</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	6274	6274
Object	fputc	fputc

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method spell\_add\_word(

```
....  
6274.                fputc('#', fd);
```

### Improper Resource Access Authorization\Path 11:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=618">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=618</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	6594	6594
Object	fputc	fputc

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_spell\_prefcond(FILE \*fd, garray\_T \*gap, size\_t \*fwv)

```
....  
6594.                fputc(len, fd);
```

#### Improper Resource Access Authorization\Path 12:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=619>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	6600	6600
Object	fputc	fputc

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_spell\_prefcond(FILE \*fd, garray\_T \*gap, size\_t \*fwv)

```
....  
6600.                fputc(0, fd);
```

#### Improper Resource Access Authorization\Path 13:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=620>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	4947	4947

Object	fputc	fputc
--------	-------	-------

## Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
4947.          fputc(128, fd);          // <charflagslen>
```

**Improper Resource Access Authorization\Path 14:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=621>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	4955	4955
Object	fputc	fputc

## Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
4955.          fputc(flags, fd);          // <charflags>
```

**Improper Resource Access Authorization\Path 15:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=622>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	6274	6274
Object	fputc	fputc

## Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method spell\_add\_word(

```
....  
6274.                fputc('#', fd);
```

#### Improper Resource Access Authorization\Path 16:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=623">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=623</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	6594	6594
Object	fputc	fputc

##### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_spell\_prefcond(FILE \*fd, garray\_T \*gap, size\_t \*fwv)

```
....  
6594.                fputc(len, fd);
```

#### Improper Resource Access Authorization\Path 17:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=624">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=624</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	6600	6600
Object	fputc	fputc

##### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_spell\_prefcond(FILE \*fd, garray\_T \*gap, size\_t \*fwv)

```
....  
6600.                fputc(0, fd);
```

#### Improper Resource Access Authorization\Path 18:

Severity	Low
----------	-----

Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=625">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=625</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	4882	4882
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
4882.          fvw &= fwrite(VIMSPELLMAGIC, VIMSPELLMAGICL, (size_t)1, fd);
```

### Improper Resource Access Authorization\Path 19:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=626">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=626</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	4901	4901
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
4901.          fvw &= fwrite(spin->si_info, (size_t)i, (size_t)1, fd); //  
<infotext>
```

### Improper Resource Access Authorization\Path 20:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=627">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=627</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	4912	4912
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
4912.          fwv &= fwrite(spin->si_region_name, (size_t)1, (size_t)1,  
fd);
```

#### Improper Resource Access Authorization\Path 21:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=628>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	4959	4959
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
4959.          fwv &= fwrite(folchars, (size_t)1, (size_t)1, fd); //  
<folchars>
```

#### Improper Resource Access Authorization\Path 22:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=629>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c

Line	4970	4970
Object	fwrite	fwrite

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
4970.          fwv &= fwrite(spin->si_midword, (size_t)i, (size_t)1, fd);
```

**Improper Resource Access Authorization\Path 23:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=630>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	5058	5058
Object	fwrite	fwrite

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
5058.          fwv &= fwrite(p, 1, (size_t)1, fd);
```

**Improper Resource Access Authorization\Path 24:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=631>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	5076	5076
Object	fwrite	fwrite

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c



Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
5076.          fvw &= fwrite(spin->si_sofofr, 1, (size_t)1, fd); //  
<sofofrom>
```

#### Improper Resource Access Authorization\Path 25:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=632>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	5080	5080
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
5080.          fvw &= fwrite(spin->si_sofoto, 1, (size_t)1, fd); //  
<sofoto>
```

#### Improper Resource Access Authorization\Path 26:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=633>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	5105	5105
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
.....  
5105.                fvw &= fwrite(hi->hi_key, (size_t)1, (size_t)1,  
fd);
```

#### Improper Resource Access Authorization\Path 27:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=634">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=634</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	5121	5121
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
.....  
5121.                fvw &= fwrite(spin->si_map.ga_data, (size_t)1, (size_t)1,  
fd);
```

#### Improper Resource Access Authorization\Path 28:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=635">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=635</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	5187	5187
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
.....  
5187.                fvw &= fwrite(p, (size_t)STRLEN(p), (size_t)1, fd);
```

**Improper Resource Access Authorization\Path 29:**

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=636">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=636</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	5191	5191
Object	fwrite	fwrite

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
5191.          fvw &= fwrite(spin->si_compflags, (size_t)STRLEN(spin->si_compflags),
```

**Improper Resource Access Authorization\Path 30:**

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=637">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=637</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	5215	5215
Object	fwrite	fwrite

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
5215.          fvw &= fwrite(spin->si_syllable, (size_t)1, (size_t)1, fd);
```

**Improper Resource Access Authorization\Path 31:**

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=638">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=638</a>

Status	New
--------	-----

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	5816	5816
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method sug\_write(spellinfo\_T \*spin, char\_u \*fname)

```
....  
5816.      if (fwrite(VIMSUGMAGIC, VIMSUGMAGICL, (size_t)1, fd) != 1) //  
<fileID>
```

### Improper Resource Access Authorization\Path 32:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=639">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=639</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	5858	5858
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method sug\_write(spellinfo\_T \*spin, char\_u \*fname)

```
....  
5858.      if (fwrite(line, (size_t)len, (size_t)1, fd) == 0)
```

### Improper Resource Access Authorization\Path 33:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=640">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=640</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c

Line	6595	6595
Object	fwrite	fwrite

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method write\_spell\_prefcond(FILE \*fd, garray\_T \*gap, size\_t \*fwv)

```
....  
6595.                *fwv &= fwrite(p, (size_t)len, (size_t)1, fd);
```

**Improper Resource Access Authorization\Path 34:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=641>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c
Line	1011	1011
Object	fwrite	fwrite

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c  
Method check\_external\_diff(diffio\_T \*diffio)

```
....  
1011.                if (fwrite("line1\n", (size_t)6, (size_t)1, fd) != 1)
```

**Improper Resource Access Authorization\Path 35:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=642>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c
Line	1019	1019
Object	fwrite	fwrite

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c

Method check\_external\_diff(diffio\_T \*diffio)

```
....  
1019.                if (fwrite("line2\n", (size_t)6, (size_t)1, fd) != 1)
```

### Improper Resource Access Authorization\Path 36:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=643>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c
Line	1011	1011
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c

Method check\_external\_diff(diffio\_T \*diffio)

```
....  
1011.                if (fwrite("line1\n", (size_t)6, (size_t)1, fd) != 1)
```

### Improper Resource Access Authorization\Path 37:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=644>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c
Line	1019	1019
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c

Method check\_external\_diff(diffio\_T \*diffio)

```
....  
1019.                if (fwrite("line2\n", (size_t)6, (size_t)1, fd) != 1)
```

### Improper Resource Access Authorization\Path 38:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=645">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=645</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	4882	4882
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
4882.          fvw &= fwrite(VIMSPELLMAGIC, VIMSPELLMAGICL, (size_t)1, fd);
```

### Improper Resource Access Authorization\Path 39:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=646">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=646</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	4901	4901
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
4901.          fvw &= fwrite(spin->si_info, (size_t)i, (size_t)1, fd); //  
<infotext>
```

### Improper Resource Access Authorization\Path 40:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=647">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=647</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	4912	4912
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
4912.          fvw &= fwrite(spin->si_region_name, (size_t)1, (size_t)1,  
fd);
```

#### Improper Resource Access Authorization\Path 41:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=648>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	4959	4959
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
4959.          fvw &= fwrite(folchars, (size_t)1, (size_t)1, fd); //  
<folchars>
```

#### Improper Resource Access Authorization\Path 42:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=649>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c



Line	4970	4970
Object	fwrite	fwrite

## Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
4970.          fvw &= fwrite(spin->si_midword, (size_t)i, (size_t)1, fd);
```

**Improper Resource Access Authorization\Path 43:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=650>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	5058	5058
Object	fwrite	fwrite

## Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
5058.          fvw &= fwrite(p, 1, (size_t)1, fd);
```

**Improper Resource Access Authorization\Path 44:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=651>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	5076	5076
Object	fwrite	fwrite

## Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c

Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
5076.          fvw &= fwrite(spin->si_sofofr, 1, (size_t)1, fd); //  
<sofofrom>
```

#### Improper Resource Access Authorization\Path 45:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=652>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	5080	5080
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
5080.          fvw &= fwrite(spin->si_sofoto, 1, (size_t)1, fd); //  
<sofoto>
```

#### Improper Resource Access Authorization\Path 46:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=653>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	5105	5105
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
5105.          fwv &= fwrite(hi->hi_key, (size_t)1, (size_t)1,  
fd);
```

#### Improper Resource Access Authorization\Path 47:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=654">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=654</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	5121	5121
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
5121.          fwv &= fwrite(spin->si_map.ga_data, (size_t)1, (size_t)1,  
fd);
```

#### Improper Resource Access Authorization\Path 48:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=655">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=655</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	5187	5187
Object	fwrite	fwrite

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
5187.          fwv &= fwrite(p, (size_t)STRLEN(p), (size_t)1, fd);
```

**Improper Resource Access Authorization\Path 49:**

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=656">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=656</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	5191	5191
Object	fwrite	fwrite

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
5191.          fvw &= fwrite(spin->si_compflags, (size_t)STRLEN(spin->si_compflags),
```

**Improper Resource Access Authorization\Path 50:**

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=657">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=657</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	5215	5215
Object	fwrite	fwrite

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method write\_vim\_spell(spellinfo\_T \*spin, char\_u \*fname)

```
....  
5215.          fvw &= fwrite(spin->si_syllable, (size_t)1, (size_t)1, fd);
```

## Use of Sizeof On a Pointer Type

Query Path:

CPP\Cx\CPP Low Visibility\Use of Sizeof On a Pointer Type Version:1

Description

**Use of Sizeof On a Pointer Type\Path 1:**

Severity	Low
----------	-----

Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=326">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=326</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	404	404
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method skip\_expr\_concatenate(

```
....  
404.          ga_init2(gap, sizeof(char_u *), 10);
```

#### Use of Sizeof On a Pointer Type\Path 2:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=327">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=327</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	408	408
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method skip\_expr\_concatenate(

```
....  
408.          ga_init2(freegap, sizeof(char_u *), 10);
```

#### Use of Sizeof On a Pointer Type\Path 3:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=328">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=328</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	2170	2170
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method init\_evalarg(evalarg\_T \*evalarg)

```
....  
2170.      ga_init2(&evalarg->eval_tofree_ga, sizeof(char_u *), 20);
```

#### Use of Sizeof On a Pointer Type\Path 4:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=329">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=329</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c	vim@@vim-v8.2.3862-CVE-2022-1420-FP.c
Line	3304	3304
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1420-FP.c  
Method eval7t(

```
....  
3304.      ga_init2(&type_list, sizeof(type_T *), 10);
```

#### Use of Sizeof On a Pointer Type\Path 5:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=330">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=330</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	1264	1264

Object	sizeof	sizeof
--------	--------	--------

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method read\_compound(FILE \*fd, slang\_T \*slang, int len)

```
....  
1264.          ga_init2(gap, sizeof(char_u *), c);
```

#### Use of Sizeof On a Pointer Type\Path 6:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=331>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	1426	1426
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method set\_sofo(slang\_T \*lp, char\_u \*from, char\_u \*to)

```
....  
1426.          ga_init2(gap, sizeof(int *), 1);
```

#### Use of Sizeof On a Pointer Type\Path 7:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=332>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	1429	1429
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method set\_sofo(slang\_T \*lp, char\_u \*from, char\_u \*to)

```
.....  
1429.          vim_memset(gap->ga_data, 0, sizeof(int *) * 256);
```

#### Use of Sizeof On a Pointer Type\Path 8:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=333">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=333</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	4299	4299
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method getroom(

```
.....  
4299.          bl->sb_used = (bl->sb_used + sizeof(char *) - 1)
```

#### Use of Sizeof On a Pointer Type\Path 9:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=334">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=334</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	4300	4300
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method getroom(

```
.....  
4300.          & ~(sizeof(char *) - 1);
```

#### Use of Sizeof On a Pointer Type\Path 10:

Severity	Low
----------	-----



Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=335">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=335</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	5915	5915
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c

Method mkspell(

```
....  
5915.      ga_init2(&spin.si_comppat, (int)sizeof(char_u *), 20);
```

#### Use of Sizeof On a Pointer Type\Path 11:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=336">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=336</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	5916	5916
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c

Method mkspell(

```
....  
5916.      ga_init2(&spin.si_prefcond, (int)sizeof(char_u *), 50);
```

#### Use of Sizeof On a Pointer Type\Path 12:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=337">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=337</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c
Line	1174	1174
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1886-TP.c  
Method op\_yank(oparg\_T \*oap, int deleting, int mess)

```
....  
1174.      y_current->y_array = lalloc_clear(sizeof(char_u *) *  
yanklines, TRUE);
```

#### Use of Sizeof On a Pointer Type\Path 13:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=338>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c
Line	1487	1487
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1886-TP.c  
Method copy\_yank\_reg(yankreg\_T \*reg)

```
....  
1487.      sizeof(char_u *) * y_current->y_size,  
TRUE);
```

#### Use of Sizeof On a Pointer Type\Path 14:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=339>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c

Line	2950	2950
Object	sizeof	sizeof

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-1886-TP.c

Method str\_to\_reg(  
  

```
....  
2950.      pp = lalloc_clear((y_ptr->y_size + newlines) * sizeof(char_u  
*), TRUE);
```

**Use of Sizeof On a Pointer Type\Path 15:**

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=340>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	723	723
Object	sizeof	sizeof

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c

Method spell\_suggest\_list(  
  

```
....  
723.      ga_init2(gap, sizeof(char_u *), sug.su_ga.ga_len + 1);
```

**Use of Sizeof On a Pointer Type\Path 16:**

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=341>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c
Line	837	837
Object	sizeof	sizeof

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c

Method diff\_try\_update(

```
....  
837.          ga_init2(&dio->dio_diff.dout_ga, sizeof(char *), 1000);
```

#### Use of Sizeof On a Pointer Type\Path 17:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=342>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c
Line	837	837
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c

Method diff\_try\_update(

```
....  
837.          ga_init2(&dio->dio_diff.dout_ga, sizeof(char *), 1000);
```

#### Use of Sizeof On a Pointer Type\Path 18:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=343>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	404	404
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c

Method skip\_expr\_concatenate(

```
....  
404.          ga_init2(gap, sizeof(char_u *), 10);
```

#### Use of Sizeof On a Pointer Type\Path 19:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=344">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=344</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	408	408
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method skip\_expr\_concatenate(

```
....  
408.          ga_init2(freegap, sizeof(char_u *), 10);
```

#### Use of Sizeof On a Pointer Type\Path 20:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=345">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=345</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	2170	2170
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method init\_evalarg(evalarg\_T \*evalarg)

```
....  
2170.          ga_init2(&evalarg->eval_tofree_ga, sizeof(char_u *), 20);
```

#### Use of Sizeof On a Pointer Type\Path 21:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=346">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=346</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c	vim@@vim-v8.2.3862-CVE-2022-2231-FP.c
Line	3304	3304
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2231-FP.c  
Method eval7t(

```
....  
3304.          ga_init2(&type_list, sizeof(type_T *), 10);
```

#### Use of Sizeof On a Pointer Type\Path 22:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=347">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=347</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c
Line	1174	1174
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2264-TP.c  
Method op\_yank(oparg\_T \*oap, int deleting, int mess)

```
....  
1174.          y_current->y_array = lalloc_clear(sizeof(char_u *) *  
yanklines, TRUE);
```

#### Use of Sizeof On a Pointer Type\Path 23:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=348">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=348</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c
Line	1487	1487

Object	sizeof	sizeof
--------	--------	--------

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2264-TP.c

Method copy\_yank\_reg(yankreg\_T \*reg)

```
....  
1487.                                sizeof(char_u *) * y_current->y_size,  
TRUE);
```

#### Use of Sizeof On a Pointer Type\Path 24:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=349>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c
Line	2950	2950
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2264-TP.c

Method str\_to\_reg(

```
....  
2950.      pp = lalloc_clear((y_ptr->y_size + newlines) * sizeof(char_u  
) , TRUE);
```

#### Use of Sizeof On a Pointer Type\Path 25:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=350>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	1264	1264
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c

Method read\_compound(FILE \*fd, slang\_T \*slang, int len)

```
....  
1264.          ga_init2(gap, sizeof(char_u *), c);
```

#### Use of Sizeof On a Pointer Type\Path 26:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=351>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	1426	1426
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c

Method set\_sofo(slang\_T \*lp, char\_u \*from, char\_u \*to)

```
....  
1426.          ga_init2(gap, sizeof(int *), 1);
```

#### Use of Sizeof On a Pointer Type\Path 27:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=352>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	1429	1429
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c

Method set\_sofo(slang\_T \*lp, char\_u \*from, char\_u \*to)

```
....  
1429.          vim_memset(gap->ga_data, 0, sizeof(int *) * 256);
```

#### Use of Sizeof On a Pointer Type\Path 28:



Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=353">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=353</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	4299	4299
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method getroom(

```
....  
4299.          bl->sb_used = (bl->sb_used + sizeof(char *) - 1)
```

#### Use of Sizeof On a Pointer Type\Path 29:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=354">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=354</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	4300	4300
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method getroom(

```
....  
4300.                                     & ~(sizeof(char *) - 1);
```

#### Use of Sizeof On a Pointer Type\Path 30:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=355">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=355</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	5915	5915
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method mkspell(

```
.....  
5915.          ga_init2(&spin.si_comppat, (int)sizeof(char_u *), 20);
```

#### Use of Sizeof On a Pointer Type\Path 31:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=356">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=356</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	5916	5916
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method mkspell(

```
.....  
5916.          ga_init2(&spin.si_prefcond, (int)sizeof(char_u *), 50);
```

#### Use of Sizeof On a Pointer Type\Path 32:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=357">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=357</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c
Line	837	837

Object	sizeof	sizeof
--------	--------	--------

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2289-TP.c  
Method diff\_try\_update(

```
....  
837.          ga_init2(&dio->dio_diff.dout_ga, sizeof(char *), 1000);
```

**Use of Sizeof On a Pointer Type\Path 33:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=358>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c
Line	1377	1377
Object	sizeof	sizeof

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2581-TP.c  
Method cleanup\_subexpr(void)

```
....  
1377.          vim_memset(rex.reg_startp, 0, sizeof(char_u *) *  
NSUBEXP);
```

**Use of Sizeof On a Pointer Type\Path 34:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=359>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c
Line	1378	1378
Object	sizeof	sizeof

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2581-TP.c  
Method cleanup\_subexpr(void)

```
.....  
1378.                vim_memset(rex.reg_endp, 0, sizeof(char_u *) * NSUBEXP);
```

#### Use of Sizeof On a Pointer Type\Path 35:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=360">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=360</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c
Line	1398	1398
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2581-TP.c  
Method cleanup\_zsubexpr(void)

```
.....  
1398.                vim_memset(reg_startzp, 0, sizeof(char_u *) * NSUBEXP);
```

#### Use of Sizeof On a Pointer Type\Path 36:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=361">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=361</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c
Line	1399	1399
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2581-TP.c  
Method cleanup\_zsubexpr(void)

```
.....  
1399.                vim_memset(reg_endzp, 0, sizeof(char_u *) * NSUBEXP);
```

#### Use of Sizeof On a Pointer Type\Path 37:

Severity	Low
----------	-----

Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=362">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=362</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c
Line	837	837
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2598-TP.c  
Method diff\_try\_update(

```
....  
837.          ga_init2(&dio->dio_diff.dout_ga, sizeof(char *), 1000);
```

#### Use of Sizeof On a Pointer Type\Path 38:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=363">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=363</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	2402	2402
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method add\_def\_function(ufunc\_T \*ufunc)

```
....  
2402.          ga_init2(&dfunc->df_var_names, sizeof(char_u *), 10);
```

#### Use of Sizeof On a Pointer Type\Path 39:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=364">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=364</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	2485	2485
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method compile\_def\_function(

```
....  
2485.          ga_init2(&cctx.ctx_type_stack, sizeof(type_T *), 50);
```

#### Use of Sizeof On a Pointer Type\Path 40:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=365">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=365</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	3185	3185
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method set\_function\_type(ufunc\_T \*ufunc)

```
....  
3185.          ga_init2(&ufunc->uf_type_list, sizeof(type_T *), 10);
```

#### Use of Sizeof On a Pointer Type\Path 41:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=366">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=366</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	3206	3206

Object	sizeof	sizeof
--------	--------	--------

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c

Method set\_function\_type(ufunc\_T \*ufunc)

```
....
3206.                ufunc->uf_arg_types, sizeof(type_T *) *
argcount);
```

#### Use of Sizeof On a Pointer Type\Path 42:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=367>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2874-TP.c	vim@@vim-v8.2.3862-CVE-2022-2874-TP.c
Line	2402	2402
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2874-TP.c

Method add\_def\_function(ufunc\_T \*ufunc)

```
....
2402.                ga_init2(&dfunc->df_var_names, sizeof(char_u *), 10);
```

#### Use of Sizeof On a Pointer Type\Path 43:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=368>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2874-TP.c	vim@@vim-v8.2.3862-CVE-2022-2874-TP.c
Line	2485	2485
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2874-TP.c

Method compile\_def\_function(

```
....
2485.          ga_init2(&cctx.ctx_type_stack, sizeof(type_T *), 50);
```

#### Use of Sizeof On a Pointer Type\Path 44:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=369">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=369</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2874-TP.c	vim@@vim-v8.2.3862-CVE-2022-2874-TP.c
Line	3185	3185
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2874-TP.c  
Method set\_function\_type(ufunc\_T \*ufunc)

```
....
3185.          ga_init2(&ufunc->uf_type_list, sizeof(type_T *), 10);
```

#### Use of Sizeof On a Pointer Type\Path 45:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=370">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=370</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2874-TP.c	vim@@vim-v8.2.3862-CVE-2022-2874-TP.c
Line	3206	3206
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2874-TP.c  
Method set\_function\_type(ufunc\_T \*ufunc)

```
....
3206.          ufunc->uf_arg_types, sizeof(type_T *) *
          argcount);
```

#### Use of Sizeof On a Pointer Type\Path 46:



Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=371">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=371</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	198	198
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c  
Method get\_function\_args(

```
....  
198.          ga_init2(newargs, (int)sizeof(char_u *), 3);
```

#### Use of Sizeof On a Pointer Type\Path 47:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=372">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=372</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	200	200
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c  
Method get\_function\_args(

```
....  
200.          ga_init2(argtypes, (int)sizeof(char_u *), 3);
```

#### Use of Sizeof On a Pointer Type\Path 48:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=373">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=373</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	202	202
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c

Method get\_function\_args(

```
....  
202.          ga_init2(default_args, (int)sizeof(char_u *), 3);
```

#### Use of Sizeof On a Pointer Type\Path 49:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=374>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	378	378
Object	sizeof	sizeof

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c

Method parse\_argument\_types(ufunc\_T \*fp, garray\_T \*argtypes, int varargs)

```
....  
378.          ga_init2(&fp->uf_type_list, sizeof(type_T *), 10);
```

#### Use of Sizeof On a Pointer Type\Path 50:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=375>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	1126	1126

Object	sizeof	sizeof
--------	--------	--------

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c  
Method lambda\_function\_body(

```
....
1126.          ga_init2(&newlines, (int)sizeof(char_u *), 10);
```

## 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	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=581">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=581</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	6659	6659
Object	cl	cl

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method set\_map\_str(slang\_T \*lp, char\_u \*map)

```
....
6659.          b[cl] = NUL;
```

#### Unchecked Array Index\Path 2:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=582">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=582</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c

Line	1567	1567
Object	i	i

## Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c

Method mb\_str2wide(char\_u \*s)

```
....  
1567.          res[i] = NUL;
```

**Unchecked Array Index\Path 3:**

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=583>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	3774	3774
Object	cnt	cnt

## Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c

Method get\_pfxlist(

```
....  
3774.          store_afflist[cnt] = NUL;
```

**Unchecked Array Index\Path 4:**

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=584>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	3810	3810
Object	cnt	cnt

## Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c

Method get\_compflags(

```
....  
3810.      store_afflist[cnt] = NUL;
```

#### Unchecked Array Index\Path 5:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=585>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c
Line	1452	1452
Object	y_idx	y_idx

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1886-TP.c

Method yank\_copy\_line(struct block\_def \*bd, long y\_idx, int exclude\_trailing\_space)

```
....  
1452.      y_current->y_array[y_idx] = pnew;
```

#### Unchecked Array Index\Path 6:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=586>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c
Line	2698	2698
Object	len	len

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1886-TP.c

Method get\_reg\_contents(int regname, int flags)

```
....  
2698.      retval[len] = NUL;
```

#### Unchecked Array Index\Path 7:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=587">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=587</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c
Line	1835	1835
Object	idx_new	idx_new

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c  
Method diff\_copy\_entry(

```
....  
1835.      dp->df_lnum[idx_new] = dp->df_lnum[idx_orig] - off;
```

#### Unchecked Array Index\Path 8:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=588">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=588</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c	vim@@vim-v8.2.3862-CVE-2022-2208-TP.c
Line	1836	1836
Object	idx_new	idx_new

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2208-TP.c  
Method diff\_copy\_entry(

```
....  
1836.      dp->df_count[idx_new] = dp->df_count[idx_orig];
```

#### Unchecked Array Index\Path 9:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=589">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=589</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c
Line	1835	1835
Object	idx_new	idx_new

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c

Method diff\_copy\_entry(  

```
.....
1835.         dp->df_lnum[idx_new] = dp->df_lnum[idx_orig] - off;
```

**Unchecked Array Index\Path 10:**

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=590>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c	vim@@vim-v8.2.3862-CVE-2022-2210-TP.c
Line	1836	1836
Object	idx_new	idx_new

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2210-TP.c

Method diff\_copy\_entry(  

```
.....
1836.         dp->df_count[idx_new] = dp->df_count[idx_orig];
```

**Unchecked Array Index\Path 11:**

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=591>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c
Line	1452	1452

Object	y_idx	y_idx
--------	-------	-------

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2264-TP.c  
Method yank\_copy\_line(struct block\_def \*bd, long y\_idx, int exclude\_trailing\_space)

```
....  
1452.          y_current->y_array[y_idx] = pnew;
```

**Unchecked Array Index\Path 12:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=592>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c
Line	2698	2698
Object	len	len

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2264-TP.c  
Method get\_reg\_contents(int regname, int flags)

```
....  
2698.          retval[len] = NUL;
```

**Unchecked Array Index\Path 13:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=593>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	6659	6659
Object	cl	cl

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method set\_map\_str(slang\_T \*lp, char\_u \*map)



```
.....  
6659.          b[c1] = NUL;
```

#### Unchecked Array Index\Path 14:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=594">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=594</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	1567	1567
Object	i	i

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method mb\_str2wide(char\_u \*s)

```
.....  
1567.          res[i] = NUL;
```

#### Unchecked Array Index\Path 15:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=595">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=595</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	3774	3774
Object	cnt	cnt

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method get\_pfxlist(

```
.....  
3774.          store_afflist[cnt] = NUL;
```

#### Unchecked Array Index\Path 16:

Severity	Low
----------	-----

Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=596">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=596</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	3810	3810
Object	cnt	cnt

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method get\_complags(

```
....  
3810.      store_afflist[cnt] = NUL;
```

#### Unchecked Array Index\Path 17:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=597">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=597</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c
Line	1835	1835
Object	idx_new	idx_new

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2289-TP.c  
Method diff\_copy\_entry(

```
....  
1835.      dp->df_lnum[idx_new] = dp->df_lnum[idx_orig] - off;
```

#### Unchecked Array Index\Path 18:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=598">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=598</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c	vim@@vim-v8.2.3862-CVE-2022-2289-TP.c
Line	1836	1836
Object	idx_new	idx_new

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2289-TP.c  
Method diff\_copy\_entry(

```
....  
1836.         dp->df_count[idx_new] = dp->df_count[idx_orig];
```

**Unchecked Array Index\Path 19:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=599>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c
Line	2429	2429
Object	len	len

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2581-TP.c  
Method reg\_submatch(int no)

```
....  
2429.         retval[len] = '\n';
```

**Unchecked Array Index\Path 20:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=600>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c
Line	2440	2440

Object	len	len
--------	-----	-----

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2581-TP.c

Method reg\_submatch(int no)

```
....  
2440.                                retval[len] = '\n';
```

**Unchecked Array Index\Path 21:**

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=601>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c
Line	2448	2448
Object	len	len

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2581-TP.c

Method reg\_submatch(int no)

```
....  
2448.                                retval[len] = NUL;
```

**Unchecked Array Index\Path 22:**

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=602>

Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c
Line	1835	1835
Object	idx_new	idx_new

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2598-TP.c

Method diff\_copy\_entry(

```
.....
1835.          dp->df_lnum[idx_new] = dp->df_lnum[idx_orig] - off;
```

### Unchecked Array Index\Path 23:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=603">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=603</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c	vim@@vim-v8.2.3862-CVE-2022-2598-TP.c
Line	1836	1836
Object	idx_new	idx_new

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2598-TP.c  
Method diff\_copy\_entry(

```
.....
1836.          dp->df_count[idx_new] = dp->df_count[idx_orig];
```

### Unchecked Array Index\Path 24:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=604">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=604</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c	vim@@vim-v8.2.3862-CVE-2022-2862-TP.c
Line	484	484
Object	lv_idx	lv_idx

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2862-TP.c  
Method reserve\_local(

```
.....
484.          ((char_u **)dfunc->df_var_names.ga_data)[lvar->lv_idx] =
```

### Unchecked Array Index\Path 25:

Severity	Low
----------	-----

Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=605">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=605</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2874-TP.c	vim@@vim-v8.2.3862-CVE-2022-2874-TP.c
Line	484	484
Object	lv_idx	lv_idx

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2874-TP.c

Method reserve\_local(

```
....  
484.      ((char_u **)dfunc->df_var_names.ga_data)[lvar->lv_idx] =
```

#### Unchecked Array Index\Path 26:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=606">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=606</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	1601	1601
Object	len	len

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c

Method deref\_func\_name(

```
....  
1601.      name[len] = NUL;
```

#### Unchecked Array Index\Path 27:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=607">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=607</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	1604	1604
Object	len	len

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c  
Method deref\_func\_name(

```
....  
1604.                name[len] = cc;
```

## 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	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=301">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=301</a>
Status	New

The spell\_read\_aff method calls the sprintf function, at line 2186 of vim@@vim-v8.2.3862-CVE-2022-1621-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	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c	vim@@vim-v8.2.3862-CVE-2022-1621-TP.c
Line	2746	2746
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1621-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
....  
2746.                sprintf((char *)buf, "^%s", items[4]);
```

#### Unchecked Return Value\Path 2:

Severity	Low
Result State	To Verify

Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=302">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=302</a>
Status	New

The `spell_read_aff` method calls the `sprintf` function, at line 2186 of `vim@@vim-v8.2.3862-CVE-2022-1621-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	<code>vim@@vim-v8.2.3862-CVE-2022-1621-TP.c</code>	<code>vim@@vim-v8.2.3862-CVE-2022-1621-TP.c</code>
Line	2748	2748
Object	<code>sprintf</code>	<code>sprintf</code>

#### Code Snippet

File Name `vim@@vim-v8.2.3862-CVE-2022-1621-TP.c`  
Method `spell_read_aff(spellinfo_T *spin, char_u *fname)`

```
....  
2748.                                sprintf((char *)buf, "%s$", items[4]);
```

#### Unchecked Return Value\Path 3:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=303">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=303</a>
Status	New

The `spell_read_aff` method calls the `sprintf` function, at line 2186 of `vim@@vim-v8.2.3862-CVE-2022-1621-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	<code>vim@@vim-v8.2.3862-CVE-2022-1621-TP.c</code>	<code>vim@@vim-v8.2.3862-CVE-2022-1621-TP.c</code>
Line	2807	2807
Object	<code>sprintf</code>	<code>sprintf</code>

#### Code Snippet

File Name `vim@@vim-v8.2.3862-CVE-2022-1621-TP.c`  
Method `spell_read_aff(spellinfo_T *spin, char_u *fname)`

```
....  
2807.                                sprintf((char *)buf, "^%s",
```

#### Unchecked Return Value\Path 4:

Severity	Low
----------	-----



Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=304">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=304</a>
Status	New

The suggest\_trie\_walk method calls the sprintf function, at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-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	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	1370	1370
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....  
1370.                sprintf(changename[depth], "prefix");
```

#### Unchecked Return Value\Path 5:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=305">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=305</a>
Status	New

The suggest\_trie\_walk method calls the sprintf function, at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-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	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	1769	1769
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....  
1769.                sprintf(changename[depth], "%. *s-%s: split",
```

#### Unchecked Return Value\Path 6:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=306">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=306</a>
Status	New

The suggest\_trie\_walk method calls the sprintf function, at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-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	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	1772	1772
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....  
1772.                                sprintf(changename[depth], "%. *s-%s:  
compound",
```

#### Unchecked Return Value\Path 7:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=307">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=307</a>
Status	New

The suggest\_trie\_walk method calls the sprintf function, at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-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	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	1918	1918
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
.....
1918.                sprintf(changename[depth], "%.s-%s: subst %c to
%c",
```

#### Unchecked Return Value\Path 8:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=308">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=308</a>
Status	New

The suggest\_rie\_walk method calls the sprintf function, at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-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	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	1922	1922
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_rie\_walk(

```
.....
1922.                sprintf(changename[depth], "%.s-%s: accept %c",
```

#### Unchecked Return Value\Path 9:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=309">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=309</a>
Status	New

The suggest\_rie\_walk method calls the sprintf function, at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-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	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2057	2057
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....  
2057.          sprintf(changename[depth], "%.s-%s: delete %c",
```

#### Unchecked Return Value\Path 10:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=310>  
Status New

The suggest\_trie\_walk method calls the sprintf function, at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-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	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2154	2154
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....  
2154.          sprintf(changename[depth], "%.s-%s: insert %c",
```

#### Unchecked Return Value\Path 11:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=311>  
Status New

The suggest\_trie\_walk method calls the sprintf function, at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-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	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2254	2254
Object	sprintf	sprintf

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c

Method suggest\_trie\_walk(  

```
....  
2254.                sprintf(changename[depth], "%.s-%s: swap %c and %c",
```

**Unchecked Return Value\Path 12:**

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=312>

Status New

The suggest\_trie\_walk method calls the sprintf function, at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-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	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2342	2342
Object	sprintf	sprintf

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c

Method suggest\_trie\_walk(  

```
....  
2342.                sprintf(changename[depth], "%.s-%s: swap3 %c and %c",
```

**Unchecked Return Value\Path 13:**

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=313>

Status New

The suggest\_trie\_walk method calls the sprintf function, at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-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	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2410	2410
Object	sprintf	sprintf

## Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....  
2410.                sprintf(changename[depth], "%.s-%s: rotate left  
%c%c%c",
```

**Unchecked Return Value\Path 14:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=314>  
Status New

The suggest\_trie\_walk method calls the sprintf function, at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-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	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2471	2471
Object	sprintf	sprintf

## Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....  
2471.                sprintf(changename[depth], "%.s-%s: rotate right  
%c%c%c",
```

**Unchecked Return Value\Path 15:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=315>  
Status New

The suggest\_trie\_walk method calls the sprintf function, at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-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	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c

Line	2583	2583
Object	sprintf	sprintf

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c

Method suggest\_trie\_walk(  

```
....  
2583.                                sprintf(changename[depth], "%.s-%s: replace %s  
with %s",
```

**Unchecked Return Value\Path 16:**

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=316>

Status New

The spell\_read\_aff method calls the sprintf function, at line 2186 of vim@@vim-v8.2.3862-CVE-2022-2287-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	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	2746	2746
Object	sprintf	sprintf

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c

Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)  

```
....  
2746.                                sprintf((char *)buf, "^%s", items[4]);
```

**Unchecked Return Value\Path 17:**

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=317>

Status New

The spell\_read\_aff method calls the sprintf function, at line 2186 of vim@@vim-v8.2.3862-CVE-2022-2287-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	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	2748	2748
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
....  
2748.                                sprintf((char *)buf, "%s$", items[4]);
```

#### Unchecked Return Value\Path 18:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=318">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=318</a>
Status	New

The spell\_read\_aff method calls the sprintf function, at line 2186 of vim@@vim-v8.2.3862-CVE-2022-2287-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	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c	vim@@vim-v8.2.3862-CVE-2022-2287-TP.c
Line	2807	2807
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2287-TP.c  
Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
....  
2807.                                sprintf((char *)buf, "^%s",
```

#### Unchecked Return Value\Path 19:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=319">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=319</a>
Status	New

The ex\_compiler method calls the sprintf function, at line 722 of vim@@vim-v8.2.3862-CVE-2022-2816-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	vim@@vim-v8.2.3862-CVE-2022-2816-TP.c	vim@@vim-v8.2.3862-CVE-2022-2816-TP.c
Line	762	762
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2816-TP.c  
Method ex\_compiler(exarg\_T \*eap)

```
....  
762.          sprintf((char *)buf, "compiler/%s.vim", eap->arg);
```

#### Unchecked Return Value\Path 20:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=320">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=320</a>
Status	New

The get\_lambda\_name method calls the sprintf function, at line 499 of vim@@vim-v8.2.3862-CVE-2022-2889-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	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	504	504
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c  
Method get\_lambda\_name(void)

```
....  
504.          sprintf((char*)name, "<lambda>%d", ++lambda_no);
```

#### Unchecked Return Value\Path 21:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=321">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=321</a>
Status	New

The fname\_trans\_sid method calls the sprintf function, at line 1813 of vim@@vim-v8.2.3862-CVE-2022-2889-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	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	1832	1832
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c

Method fname\_trans\_sid(char\_u \*name, char\_u \*fname\_buf, char\_u \*\*tofree, int \*error)

```
....  
1832.                sprintf((char *)fname_buf + 3, "%ld_",
```

#### Unchecked Return Value\Path 22:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=322>

Status New

The call\_user\_func method calls the sprintf function, at line 2456 of vim@@vim-v8.2.3862-CVE-2022-2889-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	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	2642	2642
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c

Method call\_user\_func(

```
....  
2642.                sprintf((char *)numbuf, "%d", ai + 1);
```

#### Unchecked Return Value\Path 23:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=323>

Status New

The delete\_script\_functions method calls the sprintf function, at line 2966 of vim@@vim-v8.2.3862-CVE-2022-2889-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	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	2977	2977
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c  
Method delete\_script\_functions(int sid)

```
....  
2977.          sprintf((char *)buf + 3, "%d_", sid);
```

#### Unchecked Return Value\Path 24:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=324">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=324</a>
Status	New

The trans\_function\_name method calls the sprintf function, at line 3607 of vim@@vim-v8.2.3862-CVE-2022-2889-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	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	3811	3811
Object	sprintf	sprintf

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c  
Method trans\_function\_name(

```
....  
3811.          sprintf((char *)sid_buf, "%ld_",  
(long)current_sctx.sc_sid);
```

#### Unchecked Return Value\Path 25:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=325">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=325</a>
Status	New

The `define_function` method calls the `sprintf` function, at line 3951 of `vim@@vim-v8.2.3862-CVE-2022-2889-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	<code>vim@@vim-v8.2.3862-CVE-2022-2889-TP.c</code>	<code>vim@@vim-v8.2.3862-CVE-2022-2889-TP.c</code>
Line	4434	4434
Object	<code>sprintf</code>	<code>sprintf</code>

#### Code Snippet

File Name `vim@@vim-v8.2.3862-CVE-2022-2889-TP.c`  
Method `define_function(exarg_T *eap, char_u *name_arg)`

```
....  
4434.          sprintf(numbuf, "%d", ++func_nr);
```

## Potential Precision Problem

### Query Path:

CPP\Cx\CPP Buffer Overflow\Potential Precision Problem Version:0

### Categories

NIST SP 800-53: SI-10 Information Input Validation (P1)  
OWASP Top 10 2017: A1-Injection

### Description

#### Potential Precision Problem\Path 1:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=556">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=556</a>
Status	New

The size of the buffer used by `spell_read_aff` in `"^%s"`, at line 2186 of `vim@@vim-v8.2.3862-CVE-2022-1621-TP.c`, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that `spell_read_aff` passes to `"^%s"`, at line 2186 of `vim@@vim-v8.2.3862-CVE-2022-1621-TP.c`, to overwrite the target buffer.

	Source	Destination
File	<code>vim@@vim-v8.2.3862-CVE-2022-1621-TP.c</code>	<code>vim@@vim-v8.2.3862-CVE-2022-1621-TP.c</code>
Line	2746	2746
Object	<code>"^%s"</code>	<code>"^%s"</code>

#### Code Snippet

File Name `vim@@vim-v8.2.3862-CVE-2022-1621-TP.c`  
Method `spell_read_aff(spellinfo_T *spin, char_u *fname)`

```
.....
2746.                                sprintf((char *)buf, "%s", items[4]);
```

### Potential Precision Problem\Path 2:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=557">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=557</a>
Status	New

The size of the buffer used by `spell_read_aff` in `"%s$"`, at line 2186 of `vim@@vim-v8.2.3862-CVE-2022-1621-TP.c`, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that `spell_read_aff` passes to `"%s$"`, at line 2186 of `vim@@vim-v8.2.3862-CVE-2022-1621-TP.c`, to overwrite the target buffer.

	Source	Destination
File	<code>vim@@vim-v8.2.3862-CVE-2022-1621-TP.c</code>	<code>vim@@vim-v8.2.3862-CVE-2022-1621-TP.c</code>
Line	2748	2748
Object	<code>"%s\$"</code>	<code>"%s\$"</code>

#### Code Snippet

File Name `vim@@vim-v8.2.3862-CVE-2022-1621-TP.c`  
 Method `spell_read_aff(spellinfo_T *spin, char_u *fname)`

```
.....
2748.                                sprintf((char *)buf, "%s$", items[4]);
```

### Potential Precision Problem\Path 3:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=558">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=558</a>
Status	New

The size of the buffer used by `spell_read_aff` in `"^%s"`, at line 2186 of `vim@@vim-v8.2.3862-CVE-2022-1621-TP.c`, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that `spell_read_aff` passes to `"^%s"`, at line 2186 of `vim@@vim-v8.2.3862-CVE-2022-1621-TP.c`, to overwrite the target buffer.

	Source	Destination
File	<code>vim@@vim-v8.2.3862-CVE-2022-1621-TP.c</code>	<code>vim@@vim-v8.2.3862-CVE-2022-1621-TP.c</code>
Line	2807	2807
Object	<code>"^%s"</code>	<code>"^%s"</code>

#### Code Snippet

File Name `vim@@vim-v8.2.3862-CVE-2022-1621-TP.c`

Method spell\_read\_aff(spellinfo\_T \*spin, char\_u \*fname)

```
....  
2807.                                     sprintf((char *)buf, "^%s",
```

#### Potential Precision Problem\Path 4:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=559>

Status New

The size of the buffer used by suggest\_trie\_walk in "%.s-%s: split", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that suggest\_trie\_walk passes to "%.s-%s: split", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, to overwrite the target buffer.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	1769	1769
Object	"%.s-%s: split"	"%.s-%s: split"

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c

Method suggest\_trie\_walk(

```
....  
1769.                                     sprintf(changename[depth], "%.s-%s: split",
```

#### Potential Precision Problem\Path 5:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=560>

Status New

The size of the buffer used by suggest\_trie\_walk in "%.s-%s: compound", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that suggest\_trie\_walk passes to "%.s-%s: compound", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, to overwrite the target buffer.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	1772	1772
Object	"%.s-%s: compound"	"%.s-%s: compound"

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....  
1772.                                sprintf(changename[depth], "%.s-%s:  
compound",
```

#### Potential Precision Problem\Path 6:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=561>  
Status New

The size of the buffer used by suggest\_trie\_walk in "%.s-%s: subst %c to %c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that suggest\_trie\_walk passes to "%.s-%s: subst %c to %c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, to overwrite the target buffer.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	1918	1918
Object	"%.s-%s: subst %c to %c"	"%.s-%s: subst %c to %c"

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....  
1918.                                sprintf(changename[depth], "%.s-%s: subst %c to  
%c",
```

#### Potential Precision Problem\Path 7:

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=562>  
Status New

The size of the buffer used by suggest\_trie\_walk in "%.s-%s: accept %c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that suggest\_trie\_walk passes to "%.s-%s: accept %c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, to overwrite the target buffer.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	1922	1922

Object	"%.*s-%s: accept %c"	"%.*s-%s: accept %c"
--------	----------------------	----------------------

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....  
1922.                                sprintf(changename[depth], "%.s-%s: accept %c",
```

#### Potential Precision Problem\Path 8:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=563">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=563</a>
Status	New

The size of the buffer used by suggest\_trie\_walk in "%.s-%s: delete %c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that suggest\_trie\_walk passes to "%.s-%s: delete %c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, to overwrite the target buffer.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2057	2057
Object	"%.s-%s: delete %c"	"%.s-%s: delete %c"

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....  
2057.                                sprintf(changename[depth], "%.s-%s: delete %c",
```

#### Potential Precision Problem\Path 9:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=564">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=564</a>
Status	New

The size of the buffer used by suggest\_trie\_walk in "%.s-%s: insert %c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that suggest\_trie\_walk passes to "%.s-%s: insert %c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, to overwrite the target buffer.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c



Line	2154	2154
Object	"%.*s-%s: insert %c"	"%.*s-%s: insert %c"

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c

Method suggest\_trie\_walk(

```
....
2154.          sprintf(changename[depth], "%.s-%s: insert %c",
```

#### Potential Precision Problem\Path 10:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=565>

Status New

The size of the buffer used by suggest\_trie\_walk in "%.s-%s: swap %c and %c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that suggest\_trie\_walk passes to "%.s-%s: swap %c and %c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, to overwrite the target buffer.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2254	2254
Object	"%.s-%s: swap %c and %c"	"%.s-%s: swap %c and %c"

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c

Method suggest\_trie\_walk(

```
....
2254.          sprintf(changename[depth], "%.s-%s: swap %c and %c",
```

#### Potential Precision Problem\Path 11:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=566>

Status New

The size of the buffer used by suggest\_trie\_walk in "%.s-%s: swap3 %c and %c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that suggest\_trie\_walk passes to "%.s-%s: swap3 %c and %c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, to overwrite the target buffer.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-	vim@@vim-v8.2.3862-CVE-2022-2126-

	TP.c	TP.c
Line	2342	2342
Object	"%.*s-%s: swap3 %c and %c"	"%.*s-%s: swap3 %c and %c"

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....
2342.          sprintf(changename[depth], "%.s-%s: swap3 %c and %c",
```

#### Potential Precision Problem\Path 12:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=567">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=567</a>
Status	New

The size of the buffer used by suggest\_trie\_walk in "%.s-%s: rotate left %c%c%c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that suggest\_trie\_walk passes to "%.s-%s: rotate left %c%c%c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, to overwrite the target buffer.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2410	2410
Object	"%.s-%s: rotate left %c%c%c"	"%.s-%s: rotate left %c%c%c"

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....
2410.          sprintf(changename[depth], "%.s-%s: rotate left
%c%c%c",
```

#### Potential Precision Problem\Path 13:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=568">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=568</a>
Status	New

The size of the buffer used by suggest\_trie\_walk in "%.s-%s: rotate right %c%c%c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that suggest\_trie\_walk passes to "%.s-%s: rotate right %c%c%c", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, to overwrite the target buffer.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2471	2471
Object	"%.*s-%s: rotate right %c%c%c"	"%.*s-%s: rotate right %c%c%c"

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....  
2471.             sprintf(changename[depth], "%.s-%s: rotate right  
%c%c%c",
```

#### Potential Precision Problem\Path 14:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=569">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=569</a>
Status	New

The size of the buffer used by suggest\_trie\_walk in "%.s-%s: replace %s with %s", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that suggest\_trie\_walk passes to "%.s-%s: replace %s with %s", at line 1242 of vim@@vim-v8.2.3862-CVE-2022-2126-TP.c, to overwrite the target buffer.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	2583	2583
Object	"%.s-%s: replace %s with %s"	"%.s-%s: replace %s with %s"

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c  
Method suggest\_trie\_walk(

```
....  
2583.             sprintf(changename[depth], "%.s-%s: replace %s  
with %s",
```

#### Potential Precision Problem\Path 15:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=570">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=570</a>
Status	New

The size of the buffer used by `spell_read_aff` in `"^%s"`, at line 2186 of `vim@@vim-v8.2.3862-CVE-2022-2287-TP.c`, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that `spell_read_aff` passes to `"^%s"`, at line 2186 of `vim@@vim-v8.2.3862-CVE-2022-2287-TP.c`, to overwrite the target buffer.

	Source	Destination
File	<code>vim@@vim-v8.2.3862-CVE-2022-2287-TP.c</code>	<code>vim@@vim-v8.2.3862-CVE-2022-2287-TP.c</code>
Line	2746	2746
Object	<code>"^%s"</code>	<code>"^%s"</code>

#### Code Snippet

File Name `vim@@vim-v8.2.3862-CVE-2022-2287-TP.c`  
Method `spell_read_aff(spellinfo_T *spin, char_u *fname)`

```
....  
2746.                                sprintf((char *)buf, "^%s", items[4]);
```

#### Potential Precision Problem\Path 16:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=571">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=571</a>
Status	New

The size of the buffer used by `spell_read_aff` in `"%s$"`, at line 2186 of `vim@@vim-v8.2.3862-CVE-2022-2287-TP.c`, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that `spell_read_aff` passes to `"%s$"`, at line 2186 of `vim@@vim-v8.2.3862-CVE-2022-2287-TP.c`, to overwrite the target buffer.

	Source	Destination
File	<code>vim@@vim-v8.2.3862-CVE-2022-2287-TP.c</code>	<code>vim@@vim-v8.2.3862-CVE-2022-2287-TP.c</code>
Line	2748	2748
Object	<code>"%s\$"</code>	<code>"%s\$"</code>

#### Code Snippet

File Name `vim@@vim-v8.2.3862-CVE-2022-2287-TP.c`  
Method `spell_read_aff(spellinfo_T *spin, char_u *fname)`

```
....  
2748.                                sprintf((char *)buf, "%s$", items[4]);
```

#### Potential Precision Problem\Path 17:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=572">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=572</a>
Status	New

The size of the buffer used by `spell_read_aff` in "`^%s`", at line 2186 of `vim@@vim-v8.2.3862-CVE-2022-2287-TP.c`, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that `spell_read_aff` passes to "`^%s`", at line 2186 of `vim@@vim-v8.2.3862-CVE-2022-2287-TP.c`, to overwrite the target buffer.

	Source	Destination
File	<code>vim@@vim-v8.2.3862-CVE-2022-2287-TP.c</code>	<code>vim@@vim-v8.2.3862-CVE-2022-2287-TP.c</code>
Line	2807	2807
Object	" <code>^%s</code> "	" <code>^%s</code> "

#### Code Snippet

File Name `vim@@vim-v8.2.3862-CVE-2022-2287-TP.c`  
 Method `spell_read_aff(spellinfo_T *spin, char_u *fname)`

```
....
2807.                                     sprintf((char *)buf, "%s",
```

#### Potential Precision Problem\Path 18:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=573">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=573</a>
Status	New

The size of the buffer used by `ex_compiler` in "`compiler/%s.vim`", at line 722 of `vim@@vim-v8.2.3862-CVE-2022-2816-TP.c`, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that `ex_compiler` passes to "`compiler/%s.vim`", at line 722 of `vim@@vim-v8.2.3862-CVE-2022-2816-TP.c`, to overwrite the target buffer.

	Source	Destination
File	<code>vim@@vim-v8.2.3862-CVE-2022-2816-TP.c</code>	<code>vim@@vim-v8.2.3862-CVE-2022-2816-TP.c</code>
Line	762	762
Object	" <code>compiler/%s.vim</code> "	" <code>compiler/%s.vim</code> "

#### Code Snippet

File Name `vim@@vim-v8.2.3862-CVE-2022-2816-TP.c`  
 Method `ex_compiler(exarg_T *eap)`

```
....
762.                                     sprintf((char *)buf, "compiler/%s.vim", eap->arg);
```

## 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	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=574">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=574</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c
Line	1758	1758
Object	BinaryExpr	BinaryExpr

## Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1886-TP.c  
Method do\_put(

```
....  
1758.          coladvance_force(getviscol() + (dir == FORWARD));
```

**Arithmenic Operation On Boolean\Path 2:**

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=575">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=575</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c
Line	2179	2179
Object	BinaryExpr	BinaryExpr

## Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1886-TP.c  
Method do\_put(

```
....  
2179.          if (curbuf->b_op_start.lnum + (y_type == MCHAR) - 1 +  
nr_lines
```

**Arithmenic Operation On Boolean\Path 3:**

Severity	Low
Result State	To Verify

Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=576">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=576</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c	vim@@vim-v8.2.3862-CVE-2022-1886-TP.c
Line	2185	2185
Object	BinaryExpr	BinaryExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-1886-TP.c  
Method do\_put(

```
....  
2185.                mark_adjust(curbuf->b_op_start.lnum + (y_type ==  
MCHAR),
```

#### Arithmenic Operation On Boolean\Path 4:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=577">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=577</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c
Line	1758	1758
Object	BinaryExpr	BinaryExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2264-TP.c  
Method do\_put(

```
....  
1758.                coladvance_force(getviscol() + (dir == FORWARD));
```

#### Arithmenic Operation On Boolean\Path 5:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=578">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=578</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c
Line	2179	2179
Object	BinaryExpr	BinaryExpr

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2264-TP.c  
Method do\_put(

```
.....  
2179.             if (curbuf->b_op_start.lnum + (y_type == MCHAR) - 1 +  
nr_lines
```

**Arithmenic Operation On Boolean\Path 6:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=579>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c	vim@@vim-v8.2.3862-CVE-2022-2264-TP.c
Line	2185	2185
Object	BinaryExpr	BinaryExpr

**Code Snippet**

File Name vim@@vim-v8.2.3862-CVE-2022-2264-TP.c  
Method do\_put(

```
.....  
2185.             mark_adjust(curbuf->b_op_start.lnum + (y_type ==  
MCHAR) ,
```

**Arithmenic Operation On Boolean\Path 7:**

Severity Low  
Result State To Verify  
Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=580>  
Status New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c



Line	1331	1331
Object	BinaryExpr	BinaryExpr

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2581-TP.c  
Method reg\_match\_visual(void)

```
....
1331.          if (cols < start || cols > end - (*p_sel == 'e'))
```

## Potential Off by One Error in Loops

### Query Path:

CPP\Cx\CPP Heuristic\Potential Off by One Error in Loops Version:1

### Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.1 - Injection flaws - particularly SQL injection

NIST SP 800-53: SI-16 Memory Protection (P1)

OWASP Top 10 2017: A1-Injection

### Description

#### Potential Off by One Error in Loops\Path 1:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=381">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=381</a>
Status	New

The buffer allocated by <= in vim@@vim-v8.2.3862-CVE-2022-2125-TP.c at line 289 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c	vim@@vim-v8.2.3862-CVE-2022-2125-TP.c
Line	298	298
Object	<=	<=

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2125-TP.c  
Method tabstop\_copy(int \*oldts)

```
....
298.          for (t = 0; t <= oldts[0]; ++t)
```

#### Potential Off by One Error in Loops\Path 2:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=382">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=382</a>

Status New

The buffer allocated by <= in vim@@vim-v8.2.3862-CVE-2022-2126-TP.c at line 4301 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	4388	4388
Object	<=	<=

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c

Method spell\_edit\_score\_limit\_w(

```
....  
4388.          for (round = 0; round <= 1; ++round)
```

#### Potential Off by One Error in Loops\Path 3:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=383>

Status New

The buffer allocated by <= in vim@@vim-v8.2.3862-CVE-2022-2126-TP.c at line 4126 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c	vim@@vim-v8.2.3862-CVE-2022-2126-TP.c
Line	4203	4203
Object	<=	<=

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2126-TP.c

Method spell\_edit\_score\_limit(

```
....  
4203.          for (round = 0; round <= 1; ++round)
```

#### Potential Off by One Error in Loops\Path 4:

Severity Low

Result State To Verify

Online Results <http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&projectid=50069&pathid=384>

Status New

The buffer allocated by <= in vim@@vim-v8.2.3862-CVE-2022-2183-TP.c at line 298 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c	vim@@vim-v8.2.3862-CVE-2022-2183-TP.c
Line	298	298
Object	<=	<=

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2183-TP.c  
Method tabstop\_copy(int \*oldts)

```
....  
298.         for (t = 0; t <= oldts[0]; ++t)
```

#### Potential Off by One Error in Loops\Path 5:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=385">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=385</a>
Status	New

The buffer allocated by <= in vim@@vim-v8.2.3862-CVE-2022-2889-TP.c at line 640 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c	vim@@vim-v8.2.3862-CVE-2022-2889-TP.c
Line	640	640
Object	<=	<=

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2889-TP.c  
Method function\_using\_block\_scopes(ufunc\_T \*fp, cstack\_T \*cstack)

```
....  
640.         for (i = 0; i <= cstack->cs_idx; ++i)
```

## 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	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=665">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=665</a>
Status	New

	Source	Destination
File	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c
Line	2669	2669
Object	f	f

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2581-TP.c  
Method vim\_regcomp(char\_u \*expr\_arg, int re\_flags)

```
....  
2669.          f = fopen(BT_REGEX_DEBUG_LOG_NAME, "a");
```

## TOCTOU

Query Path:

CPP\Cx\CPP Low Visibility\TOCTOU Version:1

### Description

#### TOCTOU\Path 1:

Severity	Low
Result State	To Verify
Online Results	<a href="http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=666">http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1050080&amp;projectid=50069&amp;pathid=666</a>
Status	New

The vim\_regcomp method in vim@@vim-v8.2.3862-CVE-2022-2581-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	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c	vim@@vim-v8.2.3862-CVE-2022-2581-TP.c
Line	2669	2669
Object	fopen	fopen

#### Code Snippet

File Name vim@@vim-v8.2.3862-CVE-2022-2581-TP.c  
Method vim\_regcomp(char\_u \*expr\_arg, int re\_flags)

```
.....  
2669.          f = fopen(BT_REGEX_DEBUG_LOG_NAME, "a");
```

# Divide By Zero

## Risk

### What might happen

When a program divides a number by zero, an exception will be raised. If this exception is not handled by the application, unexpected results may occur, including crashing the application. This can be considered a DoS (Denial of Service) attack, if an external user has control of the value of the denominator or can cause this error to occur.

## Cause

### How does it happen

The program receives an unexpected value, and uses it for division without filtering, validation, or verifying that the value is not zero. The application does not explicitly handle this error or prevent division by zero from occurring.

## General Recommendations

### How to avoid it

- Before dividing by an unknown value, validate the number and explicitly ensure it does not evaluate to zero.
- Validate all untrusted input from all sources, in particular verifying that it is not zero before dividing with it.
- Verify output of methods, calculations, dictionary lookups, and so on, and ensure it is not zero before dividing with the result.
- Ensure divide-by-zero errors are caught and handled appropriately.

## Source Code Examples

### Java

#### Divide by Zero

```
public float getAverage(HttpServletRequest req) {  
    int total = Integer.parseInt(req.getParameter("total"));  
    int count = Integer.parseInt(req.getParameter("count"));  
  
    return total / count;  
}
```

### Checked Division

```
public float getAverage(HttpServletRequest req) {  
    int total = Integer.parseInt(req.getParameter("total"));  
    int count = Integer.parseInt(req.getParameter("count"));  
  
    if (count > 0)  
        return total / count;  
    else  
        return 0;  
}
```

# 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  
//..
```



# 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;  
}
```

# 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;
}
```

## 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
<a href="#">CVE-2005-3119</a>	Memory leak because function does not free() an element of a data structure.
<a href="#">CVE-2004-0427</a>	Memory leak when counter variable is not decremented.
<a href="#">CVE-2002-0574</a>	Memory leak when counter variable is not decremented.
<a href="#">CVE-2005-3181</a>	Kernel uses wrong function to release a data structure, preventing data from being properly tracked by other code.
<a href="#">CVE-2004-0222</a>	Memory leak via unknown manipulations as part of protocol test suite.
<a href="#">CVE-2001-0136</a>	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	<a href="#">Indicator of Poor Code Quality</a>	<b>Seven Pernicious Kingdoms (primary)700</b>
ChildOf	Category	399	<a href="#">Resource Management Errors</a>	<b>Development Concepts (primary)699</b>
ChildOf	Category	633	<a href="#">Weaknesses that Affect Memory</a>	<b>Resource-specific Weaknesses (primary)631</b>
ChildOf	Category	730	<a href="#">OWASP Top Ten 2004 Category A9 - Denial of Service</a>	<b>Weaknesses in OWASP Top Ten (2004) (primary)711</b>
ChildOf	Weakness Base	772	<a href="#">Missing Release of Resource after Effective</a>	<b>Research Concepts (primary)1000</b>

MemberOf	View	630	<a href="#">Lifetime Weaknesses Examined by SAMATE</a>	<b>Weaknesses Examined by SAMATE (primary) 630</b> Research Concepts1000
CanFollow	Weakness Class	390	<a href="#">Detection of Error Condition Without Action</a>	

## 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 updated White Box Definitions	MITRE	Internal
2009-10-29	CWE Content Team updated Modes of Introduction, Other Notes	MITRE	Internal
2010-02-16	CWE Content Team updated Relationships	MITRE	Internal
<b>Previous Entry Names</b>			
<b>Change Date</b>	<b>Previous Entry Name</b>		
2008-04-11	Memory Leak		
2009-05-27	Failure to Release Memory Before Removing Last Reference (aka 'Memory Leak')		

[BACK TO TOP](#)

# Use of Uninitialized 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 Uninitialized Variable

Weakness ID: 457 (Weakness Variant)

Status: Draft

## Description

Description Summary

The code uses a variable that has not been initialized, leading to unpredictable or unintended results.

Extended Description

In some languages, such as C, an uninitialized variable contains contents of previously-used memory. An attacker can sometimes control or read these contents.

## Time of Introduction

## • Implementation

## Applicable Platforms

Languages

C: (Sometimes)

C++: (Sometimes)

Perl: (Often)

All

## Common Consequences

Scope	Effect
Availability Integrity	Initial variables usually contain junk, which can not be trusted for consistency. This can lead to denial of service conditions, or modify control flow in unexpected ways. In some cases, an attacker can "pre-initialize" the variable using previous actions, which might enable code execution. This can cause a race condition if a lock variable check passes when it should not.
Authorization	Strings that are not initialized are especially dangerous, since many functions expect a null at the end -- and only at the end - of a string.

## Likelihood of Exploit

High

## Demonstrative Examples

Example 1

The following switch statement is intended to set the values of the variables aN and bN, but in the default case, the programmer has accidentally set the value of aN twice. As a result, bN will have an undefined value.

(Bad Code)

Example Language: C

```
switch (ctl) {  
case -1:  
aN = 0;  
bN = 0;  
break;  
case 0:  
aN = i;  
bN = -i;  
break;  
case 1:  
aN = i + NEXT_SZ;  
bN = i - NEXT_SZ;  
break;  
default:  
aN = 0;  
bN = 0;  
break;  
}
```

```
aN = -1;
aN = -1;
break;
}
repaint(aN, bN);
```

Most uninitialized variable issues result in general software reliability problems, but if attackers can intentionally trigger the use of an uninitialized variable, they might be able to launch a denial of service attack by crashing the program. Under the right circumstances, an attacker may be able to control the value of an uninitialized variable by affecting the values on the stack prior to the invocation of the function.

## Example 2

*Example Languages: C++ and Java*

```
int foo;
void bar() {
if (foo==0)
/.../
/..//
}
```

## Observed Examples

Reference	Description
<a href="#">CVE-2008-0081</a>	Uninitialized variable leads to code execution in popular desktop application.
<a href="#">CVE-2007-4682</a>	Crafted input triggers dereference of an uninitialized object pointer.
<a href="#">CVE-2007-3468</a>	Crafted audio file triggers crash when an uninitialized variable is used.
<a href="#">CVE-2007-2728</a>	Uninitialized random seed variable used.

## Potential Mitigations

### Phase: Implementation

Assign all variables to an initial value.

### Phase: Build and Compilation

Most compilers will complain about the use of uninitialized variables if warnings are turned on.

### Phase: Requirements

The choice could be made to use a language that is not susceptible to these issues.

### Phase: Architecture and Design

Mitigating technologies such as safe string libraries and container abstractions could be introduced.

## Other Notes

Before variables are initialized, they generally contain junk data of what was left in the memory that the variable takes up. This data is very rarely useful, and it is generally advised to pre-initialize variables or set them to their first values early. If one forgets -- in the C language -- to initialize, for example a char \*, many of the simple string libraries may often return incorrect results as they expect the null termination to be at the end of a string.

Stack variables in C and C++ are not initialized by default. Their initial values are determined by whatever happens to be in their location on the stack at the time the function is invoked. Programs should never use the value of an uninitialized variable. It is not uncommon for programmers to use an uninitialized variable in code that handles errors or other rare and exceptional circumstances. Uninitialized variable warnings can sometimes indicate the presence of a typographic error in the code.

## Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	398	<a href="#">Indicator of Poor Code Quality</a>	<b>Seven Pernicious Kingdoms (primary)700</b>
ChildOf	Weakness Base	456	<a href="#">Missing Initialization</a>	<b>Development Concepts (primary)699</b> <b>Research Concepts</b>

MemberOf	View	630	<a href="#">Weaknesses Examined by SAMATE</a>	(primary)1000 Weaknesses Examined by SAMATE (primary)630
----------	------	-----	---	---

## Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
CLASP			Uninitialized variable
7 Pernicious Kingdoms			Uninitialized Variable

## White Box Definitions

A weakness where the code path has:

1. start statement that defines variable
2. end statement that accesses the variable
3. the code path does not contain a statement that assigns value to the variable

## References

mercy. "Exploiting Uninitialized Data". Jan 2006. < <http://www.felinemenace.org/~mercy/papers/UBehavior/UBehavior.zip>>.

Microsoft Security Vulnerability Research & Defense. "MS08-014 : The Case of the Uninitialized Stack Variable Vulnerability". 2008-03-11. <<http://blogs.technet.com/swi/archive/2008/03/11/the-case-of-the-uninitialized-stack-variable-vulnerability.aspx>>.

## 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, Description, Relationships, Observed Example, Other Notes, References, Taxonomy Mappings		
2009-01-12	CWE Content Team	MITRE	Internal
	updated Common Consequences, Demonstrative Examples, Potential Mitigations		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
Previous Entry Names			
Change Date	Previous Entry Name		
2008-04-11	Uninitialized Variable		

[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

### 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());
```





# 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 function. These functions often return error codes in the result, or share other status codes with its caller. The application simply ignores this result value, losing this vital information.

---

## General Recommendations

### How to avoid it

- Always check the result of any called function that returns a value, and verify the result is an expected value.
  - Ensure the calling function responds to all possible return values.
  - Expect runtime errors and handle them gracefully. Explicitly define a mechanism for handling unexpected errors.
- 

## Source Code Examples

### CPP

#### Unchecked Memory Allocation

```
buff = (char*) malloc(size);
strncpy(buff, source, size);
```

#### Safer Memory Allocation

```
buff = (char*) malloc(size+1);
if (buff==NULL) exit(1);

strncpy(buff, source, size);
buff[size] = '\0';
```

## Use of sizeof() on a Pointer Type

**Weakness ID:** 467 (*Weakness Variant*)

**Status:** Draft

### Description

### Description Summary

The code calls sizeof() on a malloced pointer type, which always returns the wordsize/8. This can produce an unexpected result if the programmer intended to determine how much memory has been allocated.

### Time of Introduction

### Implementation

### Applicable Platforms

### Languages

C

C++

### Common Consequences

Scope	Effect
Integrity	This error can often cause one to allocate a buffer that is much smaller than what is needed, leading to resultant weaknesses such as buffer overflows.

### Likelihood of Exploit

High

### Demonstrative Examples

#### Example 1

Care should be taken to ensure sizeof returns the size of the data structure itself, and not the size of the pointer to the data structure.

In this example, sizeof(foo) returns the size of the pointer.

*(Bad Code)*

*Example Languages: C and C++*

```
double *foo;
...
foo = (double *)malloc(sizeof(foo));
```

In this example, sizeof(\*foo) returns the size of the data structure and not the size of the pointer.

*(Good Code)*

*Example Languages: C and C++*

```
double *foo;
...
foo = (double *)malloc(sizeof(*foo));
```

#### Example 2

This example defines a fixed username and password. The AuthenticateUser() function is intended to accept a username and a password from an untrusted user, and check to ensure that it matches the username and password. If the username and password match, AuthenticateUser() is intended to indicate that authentication succeeded.

*(Bad Code)*

*/\* Ignore CWE-259 (hard-coded password) and CWE-309 (use of password system for authentication) for this example. \*/*

```
char *username = "admin";
char *pass = "password";

int AuthenticateUser(char *inUser, char *inPass) {
```

```
printf("Sizeof username = %d\n", sizeof(username));
printf("Sizeof pass = %d\n", sizeof(pass));

if (strcmp(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 (! strcmp(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 `strcmp()` 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	<i>(where the weakness exists independent of other weaknesses)</i>

## Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	465	<a href="#">Pointer Issues</a>	<b>Development Concepts (primary)699</b>
ChildOf	Weakness Class	682	<a href="#">Incorrect Calculation</a>	<b>Research Concepts (primary)1000</b>
ChildOf	Category	737	<a href="#">CERT C Secure Coding Section 03 - Expressions (EXP)</a>	<b>Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734</b>
ChildOf	Category	740	<a href="#">CERT C Secure Coding Section 06 - Arrays (ARR)</a>	Weaknesses Addressed by the CERT C Secure Coding Standard734
CanPrecede	Weakness Base	131	<a href="#">Incorrect Calculation of Buffer Size</a>	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](#)

# Potential Off by One Error in Loops

## Risk

### What might happen

An off by one error may result in overwriting or over-reading of unintended memory; in most cases, this can result in unexpected behavior and even application crashes. In other cases, where allocation can be controlled by an attacker, a combination of variable assignment and an off by one error can result in execution of malicious code.

---

## Cause

### How does it happen

Often when designating variables to memory, a calculation error may occur when determining size or length that is off by one.

For example in loops, when allocating an array of size 2, its cells are counted as 0,1 - therefore, if a For loop iterator on the array is incorrectly set with the start condition `i=0` and the continuation condition `i<=2`, three cells will be accessed instead of 2, and an attempt will be made to write or read cell [2], which was not originally allocated, resulting in potential corruption of memory outside the bounds of the originally assigned array.

Another example occurs when a null-byte terminated string, in the form of a character array, is copied without its terminating null-byte. Without the null-byte, the string representation is unterminated, resulting in certain functions to over-read memory as they expect the missing null terminator.

---

## General Recommendations

### How to avoid it

- Always ensure that a given iteration boundary is correct:
    - With array iterations, consider that arrays begin with cell 0 and end with cell `n-1`, for a size `n` array.
    - With character arrays and null-byte terminated string representations, consider that the null byte is required and should not be overwritten or ignored; ensure functions in use are not vulnerable to off-by-one, specifically for instances where null-bytes are automatically appended after the buffer, instead of in place of its last character.
  - Where possible, use safe functions that manage memory and are not prone to off-by-one errors.
- 

## Source Code Examples

### CPP

#### Off-By-One in For Loop

```
int *ptr;
ptr = (int*)malloc(5 * sizeof(int));
for (int i = 0; i <= 5; i++)
{
    ptr[i] = i * 2 + 1; // ptr[5] will be set, but is out of bounds
}
```

```
}
```

### Proper Iteration in For Loop

```
int *ptr;
ptr = (int*)malloc(5 * sizeof(int));
for (int i = 0; i < 5; i++)
{
    ptr[i] = i * 2 + 1; // ptr[0-4] are well defined
}
```

### Off-By-One in strncat

```
strncat(buf, input, sizeof(buf) - strlen(buf)); // actual value should be sizeof(buf) -  
strlen(buf)-1 - this form will overwrite the terminating nullbyte
```

# 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



# Potential Precision Problem

## 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 its 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 (strlen(inputString, MAX_INPUT_SIZE) < sizeof(buffer))  
    {  
        strncpy(buffer, inputString, sizeof(buffer));  
    }  
}
```

## 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	<a href="#">Source Code</a>	<b>Development Concepts (primary)699</b>
ChildOf	Weakness Class	710	<a href="#">Coding Standards Violation</a>	<b>Research Concepts (primary)1000</b>
ParentOf	Weakness Variant	107	<a href="#">Struts: Unused Validation Form</a>	<b>Research Concepts (primary)1000</b>
ParentOf	Weakness Variant	110	<a href="#">Struts: Validator Without Form Field</a>	<b>Research Concepts (primary)1000</b>
ParentOf	Category	399	<a href="#">Resource Management Errors</a>	<b>Development Concepts (primary)699</b>
ParentOf	Weakness Base	401	<a href="#">Failure to Release Memory Before Removing Last Reference ('Memory Leak')</a>	<b>Seven Pernicious Kingdoms (primary)700</b>
ParentOf	Weakness Base	404	<a href="#">Improper Resource Shutdown or Release</a>	Development Concepts699 <b>Seven Pernicious Kingdoms (primary)700</b>
ParentOf	Weakness Variant	415	<a href="#">Double Free</a>	<b>Seven Pernicious Kingdoms (primary)700</b>
ParentOf	Weakness Base	416	<a href="#">Use After Free</a>	<b>Seven Pernicious Kingdoms (primary)700</b>
ParentOf	Weakness Variant	457	<a href="#">Use of Uninitialized Variable</a>	<b>Seven Pernicious Kingdoms (primary)700</b>
ParentOf	Weakness Base	474	<a href="#">Use of Function with Inconsistent Implementations</a>	<b>Development Concepts (primary)699</b> <b>Seven Pernicious Kingdoms (primary)700</b> <b>Research Concepts (primary)1000</b>
ParentOf	Weakness Base	475	<a href="#">Undefined Behavior for Input to API</a>	<b>Development Concepts (primary)699</b> <b>Seven Pernicious Kingdoms (primary)700</b>
ParentOf	Weakness Base	476	<a href="#">NULL Pointer Dereference</a>	<b>Development Concepts</b>

				(primary)699 Seven Pernicious Kingdoms (primary)700 Research Concepts (primary)1000
ParentOf	Weakness Base	477	<a href="#">Use of Obsolete Functions</a>	Development Concepts (primary)699 Seven Pernicious Kingdoms (primary)700 Research Concepts (primary)1000
ParentOf	Weakness Variant	478	<a href="#">Missing Default Case in Switch Statement</a>	Development Concepts (primary)699
ParentOf	Weakness Variant	479	<a href="#">Unsafe Function Call from a Signal Handler</a>	Development Concepts (primary)699
ParentOf	Weakness Variant	483	<a href="#">Incorrect Block Delimitation</a>	Development Concepts (primary)699
ParentOf	Weakness Base	484	<a href="#">Omitted Break Statement in Switch</a>	Development Concepts (primary)699 Research Concepts1000
ParentOf	Weakness Variant	546	<a href="#">Suspicious Comment</a>	Development Concepts (primary)699 Research Concepts (primary)1000
ParentOf	Weakness Variant	547	<a href="#">Use of Hard-coded, Security-relevant Constants</a>	Development Concepts (primary)699 Research Concepts (primary)1000
ParentOf	Weakness Variant	561	<a href="#">Dead Code</a>	Development Concepts (primary)699 Research Concepts (primary)1000
ParentOf	Weakness Base	562	<a href="#">Return of Stack Variable Address</a>	Development Concepts (primary)699 Research Concepts1000
ParentOf	Weakness Variant	563	<a href="#">Unused Variable</a>	Development Concepts (primary)699 Research Concepts (primary)1000
ParentOf	Category	569	<a href="#">Expression Issues</a>	Development Concepts (primary)699
ParentOf	Weakness Variant	585	<a href="#">Empty Synchronized Block</a>	Development Concepts (primary)699 Research Concepts (primary)1000
ParentOf	Weakness Variant	586	<a href="#">Explicit Call to Finalize()</a>	Development Concepts (primary)699
ParentOf	Weakness Variant	617	<a href="#">Reachable Assertion</a>	Development Concepts (primary)699
ParentOf	Weakness Base	676	<a href="#">Use of Potentially Dangerous Function</a>	Development Concepts (primary)699 Research Concepts (primary)1000
MemberOf	View	700	<a href="#">Seven Pernicious Kingdoms</a>	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 updated Time of Introduction	Cigital	External
2008-09-08	CWE Content Team updated Description, Relationships, Taxonomy Mappings	MITRE	Internal
2009-10-29	CWE Content Team updated Relationships	MITRE	Internal

### Previous Entry Names

Change Date	Previous Entry Name
2008-04-11	Code Quality

[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 `ArrayIndexOutOfBoundsException` 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 cause 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
<a href="#">CVE-2005-0369</a>	large ID in packet used as array index
<a href="#">CVE-2001-1009</a>	negative array index as argument to POP LIST command
<a href="#">CVE-2003-0721</a>	Integer signedness error leads to negative array index
<a href="#">CVE-2004-1189</a>	product does not properly track a count and a maximum number, which can lead to resultant array index overflow.
<a href="#">CVE-2007-5756</a>	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	<a href="#">Improper Input Validation</a>	<b>Development Concepts (primary)699</b> <b>Research Concepts (primary)1000</b>
ChildOf	Category	189	<a href="#">Numeric Errors</a>	Development Concepts699
ChildOf	Category	633	<a href="#">Weaknesses that Affect Memory</a>	<b>Resource-specific Weaknesses (primary)631</b>
ChildOf	Category	738	<a href="#">CERT C Secure Coding Section 04 - Integers (INT)</a>	<b>Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734</b>
ChildOf	Category	740	<a href="#">CERT C Secure Coding Section 06 - Arrays (ARR)</a>	Weaknesses Addressed by the CERT C Secure Coding Standard734
ChildOf	Category	802	<a href="#">2010 Top 25 - Risky Resource Management</a>	<b>Weaknesses in the 2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800</b>
CanPrecede	Weakness Class	119	<a href="#">Failure to Constrain Operations within the Bounds of a Memory Buffer</a>	Research Concepts1000
CanPrecede	Weakness Variant	789	<a href="#">Uncontrolled Memory Allocation</a>	Research Concepts1000
PeerOf	Weakness Base	124	<a href="#">Buffer Underwrite ('Buffer Underflow')</a>	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)
<a href="#">100</a>	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
<a href="#">CVE-2009-3168</a>	Web application does not restrict access to admin scripts, allowing authenticated users to reset administrative passwords.

<a href="#">CVE-2009-2960</a>	Web application does not restrict access to admin scripts, allowing authenticated users to modify passwords of other users.
<a href="#">CVE-2009-3597</a>	Web application stores database file under the web root with insufficient access control (CWE-219), allowing direct request.
<a href="#">CVE-2009-2282</a>	Terminal server does not check authorization for guest access.
<a href="#">CVE-2009-3230</a>	Database server does not use appropriate privileges for certain sensitive operations.
<a href="#">CVE-2009-2213</a>	Gateway uses default "Allow" configuration for its authorization settings.
<a href="#">CVE-2009-0034</a>	Chain: product does not properly interpret a configuration option for a system group, allowing users to gain privileges.
<a href="#">CVE-2008-6123</a>	Chain: SNMP product does not properly parse a configuration option for which hosts are allowed to connect, allowing unauthorized IP addresses to connect.
<a href="#">CVE-2008-5027</a>	System monitoring software allows users to bypass authorization by creating custom forms.
<a href="#">CVE-2008-7109</a>	Chain: reliance on client-side security (CWE-602) allows attackers to bypass authorization using a custom client.
<a href="#">CVE-2008-3424</a>	Chain: product does not properly handle wildcards in an authorization policy list, allowing unintended access.
<a href="#">CVE-2009-3781</a>	Content management system does not check access permissions for private files, allowing others to view those files.
<a href="#">CVE-2008-4577</a>	ACL-based protection mechanism treats negative access rights as if they are positive, allowing bypass of intended restrictions.
<a href="#">CVE-2008-6548</a>	Product does not check the ACL of a page accessed using an "include" directive, allowing attackers to read unauthorized files.
<a href="#">CVE-2007-2925</a>	Default ACL list for a DNS server does not set certain ACLs, allowing unauthorized DNS queries.
<a href="#">CVE-2006-6679</a>	Product relies on the X-Forwarded-For HTTP header for authorization, allowing unintended access by spoofing the header.
<a href="#">CVE-2005-3623</a>	OS kernel does not check for a certain privilege before setting ACLs for files.
<a href="#">CVE-2005-2801</a>	Chain: file-system code performs an incorrect comparison (CWE-697), preventing defaults ACLs from being properly applied.
<a href="#">CVE-2001-1155</a>	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	<a href="#">Security Features</a>	<b>Seven Pernicious Kingdoms (primary)700</b>
ChildOf	Weakness Class	284	<a href="#">Access Control (Authorization) Issues</a>	<b>Development Concepts (primary)699</b> <b>Research Concepts (primary)1000</b>
ChildOf	Category	721	<a href="#">OWASP Top Ten 2007 Category A10 - Failure to Restrict URL Access</a>	<b>Weaknesses in OWASP Top Ten (2007) (primary)629</b>
ChildOf	Category	723	<a href="#">OWASP Top Ten 2004 Category A2 - Broken Access Control</a>	<b>Weaknesses in OWASP Top Ten (2004) (primary)711</b>
ChildOf	Category	753	<a href="#">2009 Top 25 - Porous Defenses</a>	<b>Weaknesses in the 2009 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)750</b>
ChildOf	Category	803	<a href="#">2010 Top 25 - Porous Defenses</a>	<b>Weaknesses in the 2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800</b>
ParentOf	Weakness Variant	219	<a href="#">Sensitive Data Under Web Root</a>	<b>Research Concepts (primary)1000</b>
ParentOf	Weakness Base	551	<a href="#">Incorrect Behavior Order: Authorization Before Parsing and Canonicalization</a>	<b>Development Concepts (primary)699</b> <b>Research Concepts1000</b>
ParentOf	Weakness Class	638	<a href="#">Failure to Use Complete Mediation</a>	<b>Research Concepts1000</b>
ParentOf	Weakness Base	804	<a href="#">Guessable CAPTCHA</a>	<b>Development Concepts (primary)699</b> <b>Research Concepts (primary)1000</b>

## 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)
<a href="#">1</a>	Accessing Functionality Not Properly Constrained by ACLs	
<a href="#">13</a>	Subverting Environment Variable Values	

<a href="#">17</a>	Accessing, Modifying or Executing Executable Files
<a href="#">87</a>	Forceful Browsing
<a href="#">39</a>	Manipulating Opaque Client-based Data Tokens
<a href="#">45</a>	Buffer Overflow via Symbolic Links
<a href="#">51</a>	Poison Web Service Registry
<a href="#">59</a>	Session Credential Falsification through Prediction
<a href="#">60</a>	Reusing Session IDs (aka Session Replay)
<a href="#">77</a>	Manipulating User-Controlled Variables
<a href="#">76</a>	Manipulating Input to File System Calls
<a href="#">104</a>	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
<a href="#">CVE-2009-3482</a>	Anti-virus product sets insecure "Everyone: Full Control" permissions for files under the "Program Files" folder, allowing attackers to replace executables with Trojan horses.
<a href="#">CVE-2009-3897</a>	Product creates directories with 0777 permissions at installation, allowing users to gain privileges and access a socket used for authentication.
<a href="#">CVE-2009-3489</a>	Photo editor installs a service with an insecure security descriptor, allowing users to stop or start the service, or execute commands as SYSTEM.
<a href="#">CVE-2009-3289</a>	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.
<a href="#">CVE-2009-0115</a>	Device driver uses world-writable permissions for a socket file, allowing attackers to inject arbitrary commands.
<a href="#">CVE-2009-1073</a>	LDAP server stores a cleartext password in a world-readable file.
<a href="#">CVE-2009-0141</a>	Terminal emulator creates TTY devices with world-writable permissions, allowing an attacker to write to the terminals of other users.

<a href="#">CVE-2008-0662</a>	VPN product stores user credentials in a registry key with "Everyone: Full Control" permissions, allowing attackers to steal the credentials.
<a href="#">CVE-2008-0322</a>	Driver installs its device interface with "Everyone: Write" permissions.
<a href="#">CVE-2009-3939</a>	Driver installs a file with world-writable permissions.
<a href="#">CVE-2009-3611</a>	Product changes permissions to 0777 before deleting a backup; the permissions stay insecure for subsequent backups.
<a href="#">CVE-2007-6033</a>	Product creates a share with "Everyone: Full Control" permissions, allowing arbitrary program execution.
<a href="#">CVE-2007-5544</a>	Product uses "Everyone: Full Control" permissions for memory-mapped files (shared memory) in inter-process communication, allowing attackers to tamper with a session.
<a href="#">CVE-2005-4868</a>	Database product uses read/write permissions for everyone for its shared memory, allowing theft of credentials.
<a href="#">CVE-2004-1714</a>	Security product uses "Everyone: Full Control" permissions for its configuration files.
<a href="#">CVE-2001-0006</a>	"Everyone: Full Control" permissions assigned to a mutex allows users to disable network connectivity.
<a href="#">CVE-2002-0969</a>	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	<a href="#">Permission Issues</a>	<b>Development Concepts (primary)699</b>
ChildOf	Weakness Class	668	<a href="#">Exposure of Resource to Wrong Sphere</a>	<b>Research Concepts (primary)1000</b>
ChildOf	Category	753	<a href="#">2009 Top 25 - Porous Defenses</a>	<b>Weaknesses in the 2009 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)750</b>
ChildOf	Category	803	<a href="#">2010 Top 25 - Porous Defenses</a>	<b>Weaknesses in the 2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800</b>
RequiredBy	Compound Element: Composite	689	<a href="#">Permission Race Condition During Resource Copy</a>	Research Concepts1000
ParentOf	Weakness Variant	276	<a href="#">Incorrect Default Permissions</a>	<b>Research Concepts (primary)1000</b>
ParentOf	Weakness Variant	277	<a href="#">Insecure Inherited Permissions</a>	<b>Research Concepts (primary)1000</b>
ParentOf	Weakness Variant	278	<a href="#">Insecure Preserved Inherited Permissions</a>	<b>Research Concepts (primary)1000</b>
ParentOf	Weakness Variant	279	<a href="#">Incorrect Execution- Assigned Permissions</a>	<b>Research Concepts (primary)1000</b>
ParentOf	Weakness Base	281	<a href="#">Improper Preservation of Permissions</a>	<b>Research Concepts (primary)1000</b>

## Related Attack Patterns

CAPEC-ID	Attack Pattern Name	(CAPEC Version: 1.5)
<a href="#">232</a>	Exploitation of Privilege/Trust	
<a href="#">1</a>	Accessing Functionality Not Properly Constrained by ACLs	
<a href="#">17</a>	Accessing, Modifying or Executing Executable Files	
<a href="#">60</a>	Reusing Session IDs (aka Session Replay)	
<a href="#">61</a>	Session Fixation	
<a href="#">62</a>	Cross Site Request Forgery (aka Session Riding)	
<a href="#">122</a>	Exploitation of Authorization	
<a href="#">180</a>	Exploiting Incorrectly Configured Access Control Security Levels	
<a href="#">234</a>	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, overridden 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 these logical processes attempt to retrieve and update the resource without a timely management system, such as a lock, a Race Condition will occur.

An example for when a Race Condition occurs is a resource that may return a certain value to a process for further editing, and then updated by a second process, resulting in the original process' data no longer being valid. Once the original process edits and updates the incorrect value back into the resource, the second process' update has been overwritten and lost.

---

## General Recommendations

### How to avoid it

When sharing resources between concurrent processes across the application ensure that these resources are either thread-safe, or implement a locking mechanism to ensure expected concurrent activity.

---

## Source Code Examples

### Java Different Threads Increment and Decrement The Same Counter Repeatedly, Resulting in a Race Condition

```
public static int counter = 0;
public static void start() throws InterruptedException {
    incrementCounter ic;
    decrementCounter dc;
    while(counter == 0) {
        counter = 0;
        ic = new incrementCounter();
        dc = new decrementCounter();
        ic.start();
        dc.start();
        ic.join();
        dc.join();
    }
    System.out.println(counter); //Will stop and return either -1 or 1 due to race
    condition over counter
}

public static class incrementCounter extends Thread {
    public void run() {
        counter++;
    }
}
```

```
}

public static class decrementCounter extends Thread {
    public void run() {
        counter--;
    }
}
```

### Different Threads Increment and Decrement The Same Thread-Safe Counter Repeatedly, Never Resulting in a Race Condition

```
public static int counter = 0;
public static Object lock = new Object();

public static void start() throws InterruptedException {
    incrementCounter ic;
    decrementCounter dc;
    while(counter == 0) { // because of proper locking, this condition is never false
        counter = 0;
        ic = new incrementCounter();
        dc = new decrementCounter();
        ic.start();
        dc.start();
        ic.join();
        dc.join();
    }
    System.out.println(counter); // Never reached
}

public static class incrementCounter extends Thread {
    public void run() {
        synchronized (lock) {
            counter++;
        }
    }
}

public static class decrementCounter extends Thread {
    public void run() {
        synchronized (lock) {
            counter--;
        }
    }
}
```



## Scanned Languages

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