# vul_files_42 Scan Report

| | |
|---|---|
| Project Name | vul_files_42 |
| Scan Start | Tuesday, January 7, 2025 11:31:35 PM |
| Preset | Checkmarx Default |
| Scan Time | 03h:49m:47s |
| Lines Of Code Scanned | 299372 |
| Files Scanned | 221 |
| Report Creation Time | Wednesday, January 8, 2025 9:54:02 AM |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043 |
| Team | CxServer |
| Checkmarx Version | 8.7.0 |
| Scan Type | Full |
| Source Origin | LocalPath |
| Density | 1/100 (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 |

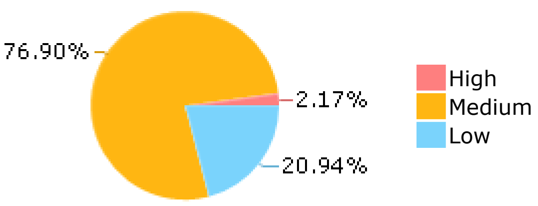
## Results Limit

Results limit per query was set to 50

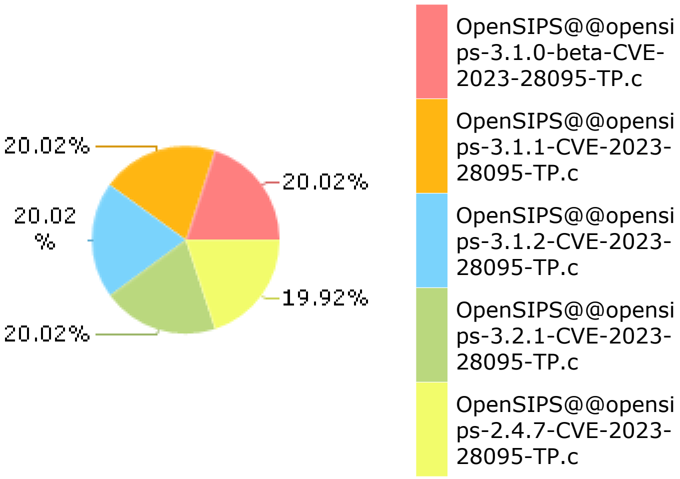
## Selected Queries

Selected queries are listed in Result Summary

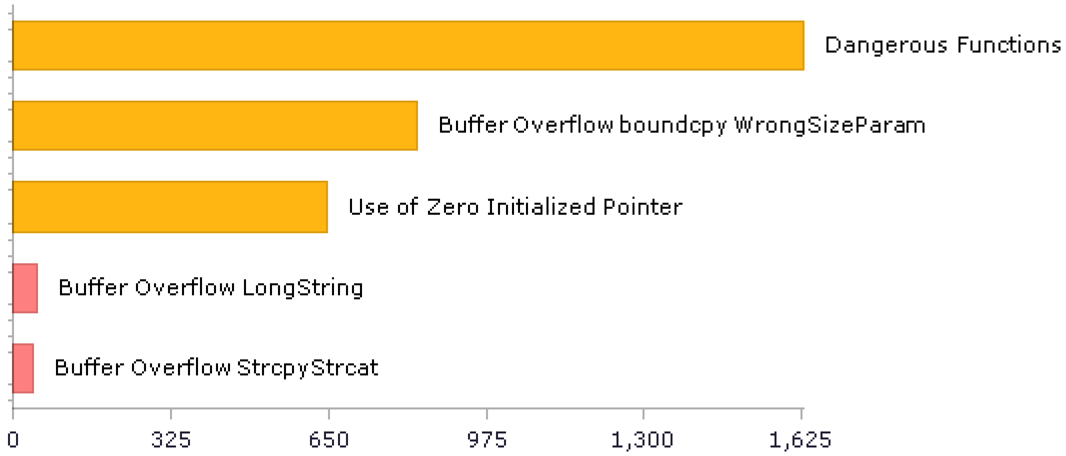
## Result Summary



## 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 | 1527 | 974 |
| A2-Broken Authentication | App. Specific | EASY | COMMON | AVERAGE | SEVERE | App. Specific | 4 | 4 |
| A3-Sensitive Data Exposure | App. Specific | AVERAGE | WIDESPREAD | AVERAGE | SEVERE | App. Specific | 19 | 19 |
| 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 | 1628 | 1628 |
| 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 | 9 | 9 |
| 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 | 1628 | 1628 |
| A10-Unvalidated Redirects and Forwards | USERS BROWSERS | AVERAGE | WIDESPREAD | DIFFICULT | MODERATE | AFFECTED DATA AND FUNCTIONS | 0 | 0 |

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

# Scan Summary - PCI DSS v3.2

| Category | Issues Found | Best Fix Locations |
|---|---|---|
| PCI DSS (3.2) - 6.5.1 - Injection flaws - particularly SQL injection | 0 | 0 |
| PCI DSS (3.2) - 6.5.2 - Buffer overflows | 928 | 848 |
| 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. | 4 | 4 |
| Audit And Accountability* | Organizations must: (i) create, protect, and retain information system audit records to the extent needed to enable the monitoring, analysis, investigation, and reporting of unlawful, unauthorized, or inappropriate information system activity; and (ii) ensure that the actions of individual information system users can be uniquely traced to those users so they can be held accountable for their actions. | 0 | 0 |
| Configuration Management | Organizations must: (i) establish and maintain baseline configurations and inventories of organizational information systems (including hardware, software, firmware, and documentation) throughout the respective system development life cycles; and (ii) establish and enforce security configuration settings for information technology products employed in organizational information systems. | 0 | 0 |
| Identification And Authentication* | Organizations must identify information system users, processes acting on behalf of users, or devices and authenticate (or verify) the identities of those users, processes, or devices, as a prerequisite to allowing access to organizational information systems. | 16 | 8 |
| 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. | 15 | 15 |
| System And Communications Protection | Organizations must: (i) monitor, control, and protect organizational communications (i.e., information transmitted or received by organizational information systems) at the external boundaries and key internal boundaries of the information systems; and (ii) employ architectural designs, software development techniques, and systems engineering principles that promote effective information security within organizational information systems. | 0 | 0 |
| System And Information Integrity | Organizations must: (i) identify, report, and correct information and information system flaws in a timely manner; (ii) provide protection from malicious code at appropriate locations within organizational information systems; and (iii) monitor information system security alerts and advisories and take appropriate actions in response. | 4 | 4 |

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

# Scan Summary - NIST SP 800-53

| Category | Issues Found | Best Fix Locations |
|---|---|---|
| AC-12 Session Termination (P2) | 0 | 0 |
| AC-3 Access Enforcement (P1) | 4 | 4 |
| 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) | 4 | 4 |
| 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)***** | 12 | 4 |
| SC-28 Protection of Information at Rest (P1) | 6 | 6 |
| SC-4 Information in Shared Resources (P1) | 9 | 9 |
| SC-5 Denial of Service Protection (P1)***** | 1294 | 313 |
| SC-8 Transmission Confidentiality and Integrity (P1) | 0 | 0 |
| SI-10 Information Input Validation (P1)***** | 207 | 127 |
| SI-11 Error Handling (P2)***** | 65 | 65 |
| SI-15 Information Output Filtering (P0) | 0 | 0 |
| SI-16 Memory Protection (P1) | 4 | 4 |

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

## Scan Summary - 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:<br>-Failing to identify the user at all when that should be required<br>-Failure to maintain the user's identity when it is required<br>-Weaknesses in session management | 0 | 0 |
| M5-Insufficient Cryptography | The code applies cryptography to a sensitive information asset. However, the cryptography is insufficient in some way. Note that anything and everything related to TLS or SSL goes in M3. Also, if the app fails to use cryptography at all when it should, that probably belongs in M2. This category is for issues where cryptography was attempted, but it wasnt done correctly. | 0 | 0 |
| M6-Insecure Authorization | This is a category to capture any failures in authorization (e.g., authorization decisions in the client side, forced browsing, etc.). It is distinct from authentication issues (e.g., device enrolment, user identification, etc.).<br>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

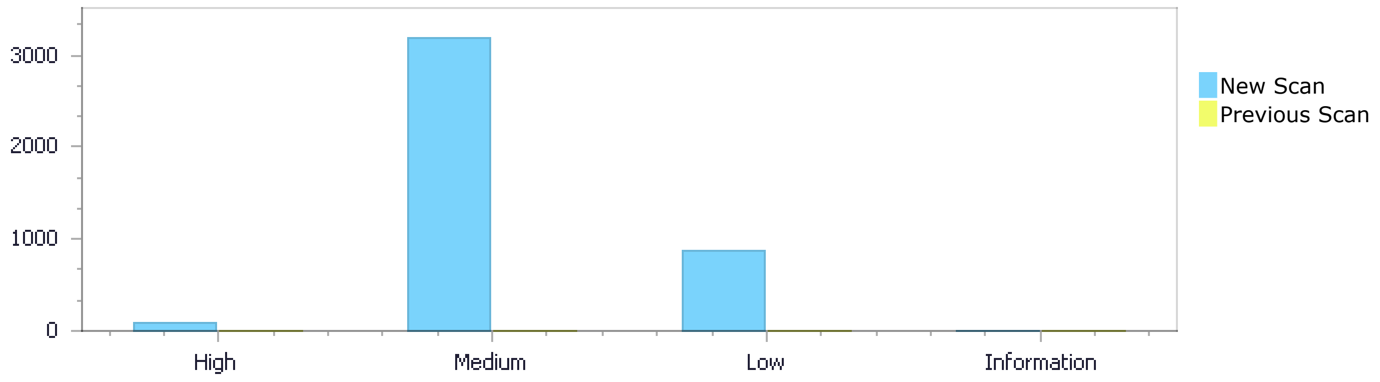
| Category | Issues Found | Best Fix Locations |
|---|---|---|
| Must audit | 0 | 0 |
| Check | 0 | 0 |
| Optional | 0 | 0 |

## Results Distribution By Status

First scan of the project

| | High | Medium | Low | Information | Total |
|---|---|---|---|---|---|
| New Issues | 90 | 3,195 | 870 | 0 | 4,155 |
| Recurrent Issues | 0 | 0 | 0 | 0 | 0 |
| Total | 90 | 3,195 | 870 | 0 | 4,155 |

| | | | | | |
|---|---|---|---|---|---|
| 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 | 90 | 3,195 | 870 | 0 | 4,155 |
| Urgent | 0 | 0 | 0 | 0 | 0 |
| Proposed Not Exploitable | 0 | 0 | 0 | 0 | 0 |
| Total | 90 | 3,195 | 870 | 0 | 4,155 |

## Result Summary

| Vulnerability Type | Occurrences | Severity |
|---|---|---|
| Buffer Overflow LongString | 48 | High |
| Buffer Overflow StrcpyStrcat | 42 | High |
| Dangerous Functions | 1628 | Medium |
| Buffer Overflow boundcpy WrongSizeParam | 834 | Medium |
| Use of Zero Initialized Pointer | 648 | Medium |

| | | |
|---|---|---|
| Use of Uninitialized Pointer | 30 | Medium |
| Memory Leak | 13 | Medium |
| Heap Inspection | 9 | Medium |
| Wrong Size t Allocation | 9 | Medium |
| MemoryFree on StackVariable | 6 | Medium |
| Wrong Memory Allocation | 6 | Medium |
| Double Free | 4 | Medium |
| Integer Overflow | 4 | Medium |
| Use of Hard coded Cryptographic Key | 4 | Medium |
| NULL Pointer Dereference | 603 | Low |
| Unchecked Array Index | 107 | Low |
| Use of Sizeof On a Pointer Type | 72 | Low |
| Unchecked Return Value | 65 | Low |
| Reliance on DNS Lookups in a Decision | 12 | Low |
| Use of Insufficiently Random Values | 6 | Low |
| Incorrect Permission Assignment For Critical Resources | 4 | Low |
| Inconsistent Implementations | 1 | Low |

## 10 Most Vulnerable Files

### High and Medium Vulnerabilities

| File Name | Issues Found |
|---|---|
| OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c | 203 |
| OpenSIPS@@opensips-3.1.1-CVE-2023-28095-TP.c | 203 |
| OpenSIPS@@opensips-3.1.2-CVE-2023-28095-TP.c | 203 |
| OpenSIPS@@opensips-3.2.1-CVE-2023-28095-TP.c | 203 |
| OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c | 202 |
| OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | 81 |
| OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | 81 |
| OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | 81 |
| OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c | 81 |
| OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c | 81 |

# Scan Results Details

## Buffer Overflow LongString

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

### Categories

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

*Description*

**Buffer Overflow LongString\Path 1:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=1 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.      name = "127.0.0.1";
....
743.      if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

**Buffer Overflow LongString\Path 2:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow

attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name)        /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
743.        if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

**Buffer Overflow LongString\Path 3:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name)        /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
743.        if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

**Buffer Overflow LongString\Path 4:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
743.        if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

**Buffer Overflow LongString\Path 5:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=5 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
746.        if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

**Buffer Overflow LongString\Path 6:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=6](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=6) |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
746.        if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

**Buffer Overflow LongString\Path 7:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=7](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=7) |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet

| | |
|---|---|
| File Name | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Method | httpGetHostByName(const char *name)        /* I - Hostname or IP address */ |

```
....
698.       name = "127.0.0.1";
....
746.       if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

### Buffer Overflow LongString\Path 8:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=8 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet

| | |
|---|---|
| File Name | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Method | httpGetHostByName(const char *name)        /* I - Hostname or IP address */ |

```
....
698.       name = "127.0.0.1";
....
746.       if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

### Buffer Overflow LongString\Path 9:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=9 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024- | OpenPrinting@@cups-v2.4.2-CVE-2024- |

| | | |
|---|---|---|
| | 35235-TP.c | 35235-TP.c |
| Line | 698 | 749 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
749.        cg->ip_addr = htonl(((((((unsigned)ip[0] << 8) |
(unsigned)ip[1]) << 8) |
```

**Buffer Overflow LongString\Path 10:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=10 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 698 | 749 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
749.        cg->ip_addr = htonl(((((((unsigned)ip[0] << 8) |
(unsigned)ip[1]) << 8) |
```

**Buffer Overflow LongString\Path 11:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=11 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 698 | 751 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
751.                              (unsigned)ip[3]));
```

### Buffer Overflow LongString\Path 12:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=12 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 698 | 750 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
750.                               (unsigned)ip[2]) << 8) |
```

### Buffer Overflow LongString\Path 13:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=13 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 698 | 749 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
749.        cg->ip_addr = htonl((ip[0] << 24) | (ip[1] << 16) | (ip[2] <<
8) | ip[3]);
```

**Buffer Overflow LongString\Path 14:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=14 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 698 | 749 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
749.        cg->ip_addr = htonl((ip[0] << 24) | (ip[1] << 16) | (ip[2] <<
8) | ip[3]);
```

### Buffer Overflow LongString\Path 15:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=15 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 698 | 749 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
749.        cg->ip_addr = htonl((ip[0] << 24) | (ip[1] << 16) | (ip[2] <<
8) | ip[3]);
```

### Buffer Overflow LongString\Path 16:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=16 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |

| | | |
|---|---|---|
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.      name = "127.0.0.1";
....
743.      if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

### Buffer Overflow LongString\Path 17:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=17 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.      name = "127.0.0.1";
....
743.      if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

### Buffer Overflow LongString\Path 18:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=18 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow

attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.      name = "127.0.0.1";
....
743.      if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

**Buffer Overflow LongString\Path 19:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=19 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.      name = "127.0.0.1";
....
746.      if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

**Buffer Overflow LongString\Path 20:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20 |

| | |
|---|---|
| | 043&pathid=20 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
746.        if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

**Buffer Overflow LongString\Path 21:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=21 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
746.        if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

### Buffer Overflow LongString\Path 22:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=22 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet

File Name OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c

Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
746.        if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

### Buffer Overflow LongString\Path 23:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=23 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 698 | 749 |
| Object | "127.0.0.1" | ip |

Code Snippet

File Name OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c

Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
749.        cg->ip_addr = htonl((ip[0] << 24) | (ip[1] << 16) | (ip[2] <<
8) | ip[3]);
```

### Buffer Overflow LongString\Path 24:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=24](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=24) |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name)  /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
743.        if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

### Buffer Overflow LongString\Path 25:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=25](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=25) |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |

| | | |
|---|---|---|
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
743.        if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

**Buffer Overflow LongString\Path 26:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=26 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
743.        if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

**Buffer Overflow LongString\Path 27:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=27 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow

attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
743.        if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

**Buffer Overflow LongString\Path 28:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=28 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
746.        if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

**Buffer Overflow LongString\Path 29:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20 |

| | |
|---|---|
| | 043&pathid=29 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
746.        if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

**Buffer Overflow LongString\Path 30:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=30 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
746.        if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

### Buffer Overflow LongString\Path 31:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=31 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
746.        if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

### Buffer Overflow LongString\Path 32:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=32 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 698 | 749 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
749.        cg->ip_addr = htonl((ip[0] << 24) | (ip[1] << 16) | (ip[2] <<
8) | ip[3]);
```

### Buffer Overflow LongString\Path 33:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=33 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 698 | 749 |
| Object | "127.0.0.1" | ip |

Code Snippet

File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c

Method httpGetHostByName(const char *name)        /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
749.        cg->ip_addr = htonl((ip[0] << 24) | (ip[1] << 16) | (ip[2] <<
8) | ip[3]);
```

### Buffer Overflow LongString\Path 34:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=34 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |

| | | |
|---|---|---|
| Line | 698 | 749 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
749.        cg->ip_addr = htonl((ip[0] << 24) | (ip[1] << 16) | (ip[2] <<
8) | ip[3]);
```

**Buffer Overflow LongString\Path 35:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=35 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 698 | 749 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
749.        cg->ip_addr = htonl((ip[0] << 24) | (ip[1] << 16) | (ip[2] <<
8) | ip[3]);
```

**Buffer Overflow LongString\Path 36:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=36 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow

attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
743.        if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

**Buffer Overflow LongString\Path 37:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=37 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
743.        if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

**Buffer Overflow LongString\Path 38:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=38 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
743.        if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

**Buffer Overflow LongString\Path 39:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=39 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
743.        if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

### Buffer Overflow LongString\Path 40:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=40 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 698 | 743 |
| Object | "127.0.0.1" | ip |

Code Snippet

| | |
|---|---|
| File Name | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Method | httpGetHostByName(const char *name)        /* I - Hostname or IP address */ |

```
....
698.        name = "127.0.0.1";
....
743.        if (sscanf(name, "%u.%u.%u.%u", ip, ip + 1, ip + 2, ip + 3) !=
4)
```

### Buffer Overflow LongString\Path 41:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=41 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |

| | | |
|---|---|---|
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.      name = "127.0.0.1";
....
746.      if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

**Buffer Overflow LongString\Path 42:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=42 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.      name = "127.0.0.1";
....
746.      if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

**Buffer Overflow LongString\Path 43:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=43 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
746.        if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

**Buffer Overflow LongString\Path 44:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=44](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=44) |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 698 | 746 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
746.        if (ip[0] > 255 || ip[1] > 255 || ip[2] > 255 || ip[3] > 255)
```

**Buffer Overflow LongString\Path 45:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=45](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=45) |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 698 | 749 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
749.        cg->ip_addr = htonl((ip[0] << 24) | (ip[1] << 16) | (ip[2] <<
8) | ip[3]);
```

**Buffer Overflow LongString\Path 46:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=46 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 698 | 749 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
749.        cg->ip_addr = htonl((ip[0] << 24) | (ip[1] << 16) | (ip[2] <<
8) | ip[3]);
```

**Buffer Overflow LongString\Path 47:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=47 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 698 | 749 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
749.        cg->ip_addr = htonl((ip[0] << 24) | (ip[1] << 16) | (ip[2] <<
8) | ip[3]);
```

**Buffer Overflow LongString\Path 48:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=48 |
| Status | New |

The size of the buffer used by httpGetHostByName in ip, at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that httpGetHostByName passes to "127.0.0.1", at line 682 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 698 | 749 |
| Object | "127.0.0.1" | ip |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpGetHostByName(const char *name) /* I - Hostname or IP address */

```
....
698.        name = "127.0.0.1";
....
749.        cg->ip_addr = htonl((ip[0] << 24) | (ip[1] << 16) | (ip[2] <<
8) | ip[3]);
```

## Buffer Overflow StrcpyStrcat

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

### Categories

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

*Description*

**Buffer Overflow StrcpyStrcat\Path 1:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=49 |
| Status | New |

The size of the buffer used by *print_string_ptr in str, at line 670 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 670 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 670 | 727 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
670.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
727.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 2:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=50 |
| Status | New |

The size of the buffer used by *print_string_ptr in str, at line 670 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 670 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 670 | 727 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
670.  static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)
....
727.            strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 3:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=51 |
| Status | New |

The size of the buffer used by *print_string_ptr in ptr2, at line 670 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 670 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 670 | 727 |
| Object | str | ptr2 |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
670.  static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)
....
727.            strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 4:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=52 |
| Status | New |

The size of the buffer used by *print_object in ptr, at line 1444 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 670 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 670 | 1693 |
| Object | str | ptr |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
670.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
```

File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c
Method static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1693.                strcpy((char*)ptr, (char*)entries[i]);
```

**Buffer Overflow StrcpyStrcat\Path 5:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=53 |
| Status | New |

The size of the buffer used by *print_object in ptr, at line 1444 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 670 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 670 | 1693 |

| | | |
|---|---|---|
| Object | str | ptr |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
670.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
```

File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c
Method static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1693.                 strcpy((char*)ptr, (char*)entries[i]);
```

**Buffer Overflow StrcpyStrcat\Path 6:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=54 |
| Status | New |

The size of the buffer used by *print_string_ptr in ptr2, at line 670 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 670 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 670 | 727 |
| Object | str | ptr2 |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
670.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
727.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 7:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=55 |
| Status | New |

The size of the buffer used by *print_string_ptr in out, at line 670 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 670 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 670 | 694 |
| Object | str | out |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
670.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
694.           strcpy((char*)out, "\"\"");
```

**Buffer Overflow StrcpyStrcat\Path 8:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=56 |
| Status | New |

The size of the buffer used by *print_string_ptr in str, at line 693 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 693 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 693 | 750 |
| Object | str | str |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
693.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
750.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 9:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=57 |
| Status | New |

The size of the buffer used by *print_string_ptr in str, at line 693 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 693 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 693 | 750 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
693.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
750.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 10:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=58 |
| Status | New |

The size of the buffer used by *print_string_ptr in ptr2, at line 693 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 693 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |

| | | |
|---|---|---|
| Line | 693 | 750 |
| Object | str | ptr2 |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
693.  static unsigned char *print_string_ptr(const unsigned char *str, 
printbuffer *p)
....
750.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 11:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=59 |
| Status | New |

The size of the buffer used by *print_object in ptr, at line 1493 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 693 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 693 | 1742 |
| Object | str | ptr |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
693.  static unsigned char *print_string_ptr(const unsigned char *str, 
printbuffer *p)
```

File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c
Method static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1742.            strcpy((char*)ptr, (char*)entries[i]);
```

**Buffer Overflow StrcpyStrcat\Path 12:**

| | |
|---|---|
| Severity | High |

| | |
|---|---|
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=60 |
| Status | New |

The size of the buffer used by *print_object in ptr, at line 1493 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 693 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 693 | 1742 |
| Object | str | ptr |

Code Snippet

File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c

Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
693.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
```

File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c

Method static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1742.                 strcpy((char*)ptr, (char*)entries[i]);
```

### Buffer Overflow StrcpyStrcat\Path 13:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=61 |
| Status | New |

The size of the buffer used by *print_string_ptr in ptr2, at line 693 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 693 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 693 | 750 |
| Object | str | ptr2 |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
693.  static unsigned char *print_string_ptr(const unsigned char *str, 
printbuffer *p)
....
750.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 14:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=62 |
| Status | New |

The size of the buffer used by *print_string_ptr in out, at line 693 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 693 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 693 | 717 |
| Object | str | out |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
693.  static unsigned char *print_string_ptr(const unsigned char *str, 
printbuffer *p)
....
717.          strcpy((char*)out, "\"\"");
```

**Buffer Overflow StrcpyStrcat\Path 15:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=63 |
| Status | New |

The size of the buffer used by *print_string_ptr in str, at line 694 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | str |

Code Snippet
File Name	OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c
Method	static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
751.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 16:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=64 |
| Status | New |

The size of the buffer used by *print_string_ptr in str, at line 694 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | str |

Code Snippet
File Name	OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c
Method	static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
751.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 17:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=65 |

| | |
|---|---|
| Status | New |

The size of the buffer used by *print_string_ptr in ptr2, at line 694 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | ptr2 |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
751.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 18:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=66 |
| Status | New |

The size of the buffer used by *print_object in ptr, at line 1494 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 694 | 1743 |
| Object | str | ptr |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
```

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p) |

```
....
1743.               strcpy((char*)ptr, (char*)entries[i]);
```

**Buffer Overflow StrcpyStrcat\Path 19:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=67](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=67) |
| Status | New |

The size of the buffer used by *print_object in ptr, at line 1494 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 694 | 1743 |
| Object | str | ptr |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
```

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p) |

```
....
1743.               strcpy((char*)ptr, (char*)entries[i]);
```

**Buffer Overflow StrcpyStrcat\Path 20:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=68](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=68) |
| Status | New |

The size of the buffer used by *print_string_ptr in ptr2, at line 694 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | ptr2 |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
751.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 21:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=69 |
| Status | New |

The size of the buffer used by *print_string_ptr in out, at line 694 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 694 | 718 |
| Object | str | out |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
718.          strcpy((char*)out, "\"\"");
```

**Buffer Overflow StrcpyStrcat\Path 22:**

| | |
|---|---|
| Severity | High |

| | |
|---|---|
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=70 |
| Status | New |

The size of the buffer used by *print_string_ptr in str, at line 694 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | str |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
751.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 23:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=71 |
| Status | New |

The size of the buffer used by *print_string_ptr in str, at line 694 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | str |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
751.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 24:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=72 |
| Status | New |

The size of the buffer used by *print_string_ptr in ptr2, at line 694 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | ptr2 |

Code Snippet
File Name OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
751.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 25:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=73 |
| Status | New |

The size of the buffer used by *print_object in ptr, at line 1494 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |

| | | |
|---|---|---|
| Line | 694 | 1743 |
| Object | str | ptr |

Code Snippet
File Name OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
```

File Name OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c

Method static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1743.              strcpy((char*)ptr, (char*)entries[i]);
```

**Buffer Overflow StrcpyStrcat\Path 26:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=74 |
| Status | New |

The size of the buffer used by *print_object in ptr, at line 1494 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 694 | 1743 |
| Object | str | ptr |

Code Snippet
File Name OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
```

File Name OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c

Method static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1743.                strcpy((char*)ptr, (char*)entries[i]);
```

### Buffer Overflow StrcpyStrcat\Path 27:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=75 |
| Status | New |

The size of the buffer used by *print_string_ptr in ptr2, at line 694 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | ptr2 |

Code Snippet

File Name OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c

Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
751.          strcpy((char*)ptr2, (const char*)str);
```

### Buffer Overflow StrcpyStrcat\Path 28:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=76 |
| Status | New |

The size of the buffer used by *print_string_ptr in out, at line 694 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 694 | 718 |
| Object | str | out |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str, 
printbuffer *p)
....
718.           strcpy((char*)out, "\"\"");
```

### Buffer Overflow StrcpyStrcat\Path 29:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=77 |
| Status | New |

The size of the buffer used by *print_string_ptr in str, at line 694 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | str |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str, 
printbuffer *p)
....
751.           strcpy((char*)ptr2, (const char*)str);
```

### Buffer Overflow StrcpyStrcat\Path 30:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=78 |
| Status | New |

The size of the buffer used by *print_string_ptr in str, at line 694 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str, 
printbuffer *p)
....
751.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 31:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=79 |
| Status | New |

The size of the buffer used by *print_string_ptr in ptr2, at line 694 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | ptr2 |

Code Snippet
File Name OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str, 
printbuffer *p)
....
751.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 32:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=80 |

| | |
|---|---|
| Status | New |

The size of the buffer used by *print_object in ptr, at line 1494 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Line | 694 | 1743 |
| Object | str | ptr |

Code Snippet

File Name OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c

Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
```

▾

File Name OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c

Method static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1743.               strcpy((char*)ptr, (char*)entries[i]);
```

**Buffer Overflow StrcpyStrcat\Path 33:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=81 |
| Status | New |

The size of the buffer used by *print_object in ptr, at line 1494 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Line | 694 | 1743 |
| Object | str | ptr |

Code Snippet

File Name OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c

Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
```

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p) |

```
....
1743.                strcpy((char*)ptr, (char*)entries[i]);
```

**Buffer Overflow StrcpyStrcat\Path 34:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=82](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=82) |
| Status | New |

The size of the buffer used by *print_string_ptr in ptr2, at line 694 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | ptr2 |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
751.            strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 35:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=83](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=83) |
| Status | New |

The size of the buffer used by *print_string_ptr in out, at line 694 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Line | 694 | 718 |
| Object | str | out |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
718.            strcpy((char*)out, "\"\"");
```

### Buffer Overflow StrcpyStrcat\Path 36:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=84 |
| Status | New |

The size of the buffer used by *print_string_ptr in str, at line 694 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | str |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
751.            strcpy((char*)ptr2, (const char*)str);
```

### Buffer Overflow StrcpyStrcat\Path 37:

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=85 |
| Status | New |

The size of the buffer used by *print_string_ptr in str, at line 694 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | str |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)
....
751.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 38:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=86 |
| Status | New |

The size of the buffer used by *print_string_ptr in ptr2, at line 694 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | ptr2 |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
751.          strcpy((char*)ptr2, (const char*)str);
```

### Buffer Overflow StrcpyStrcat\Path 39:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=87 |
| Status | New |

The size of the buffer used by *print_object in ptr, at line 1494 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c |
| Line | 694 | 1743 |
| Object | str | ptr |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
```

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p) |

```
....
1743.            strcpy((char*)ptr, (char*)entries[i]);
```

### Buffer Overflow StrcpyStrcat\Path 40:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=88 |
| Status | New |

The size of the buffer used by *print_object in ptr, at line 1494 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c |
| Line | 694 | 1743 |
| Object | str | ptr |

Code Snippet
File Name OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str, 
printbuffer *p)
```

File Name OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c

Method static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1743.              strcpy((char*)ptr, (char*)entries[i]);
```

**Buffer Overflow StrcpyStrcat\Path 41:**

Severity High
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=89
Status New

The size of the buffer used by *print_string_ptr in ptr2, at line 694 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c |
| Line | 694 | 751 |
| Object | str | ptr2 |

Code Snippet
File Name OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
751.          strcpy((char*)ptr2, (const char*)str);
```

**Buffer Overflow StrcpyStrcat\Path 42:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=90 |
| Status | New |

The size of the buffer used by *print_string_ptr in out, at line 694 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *print_string_ptr passes to str, at line 694 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c |
| Line | 694 | 718 |
| Object | str | out |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
694.  static unsigned char *print_string_ptr(const unsigned char *str,
printbuffer *p)
....
718.          strcpy((char*)out, "\"\"");
```

## Dangerous Functions

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

### Categories

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

*Description*

**Dangerous Functions\Path 1:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=957 |
| Status | New |

The dangerous function, memcpy, was found in use at line 74 in openrazer@@openrazer-v2.7.0-CVE-2022-23467-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v2.7.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v2.7.0-CVE-2022-23467-TP.c |
| Line | 104 | 104 |
| Object | memcpy | memcpy |

Code Snippet
File Name openrazer@@openrazer-v2.7.0-CVE-2022-23467-TP.c
Method int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max)

```
....
104.        memcpy(response_report, buf, sizeof(struct razer_report));
```

**Dangerous Functions\Path 2:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=958 |
| Status | New |

The dangerous function, memcpy, was found in use at line 74 in openrazer@@openrazer-v2.8.0-CVE-2022-23467-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v2.8.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v2.8.0-CVE-2022-23467-TP.c |
| Line | 104 | 104 |
| Object | memcpy | memcpy |

Code Snippet
File Name openrazer@@openrazer-v2.8.0-CVE-2022-23467-TP.c
Method int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max)

```
....
104.        memcpy(response_report, buf, sizeof(struct razer_report));
```

**Dangerous Functions\Path 3:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=959 |
| Status | New |

The dangerous function, memcpy, was found in use at line 74 in openrazer@@openrazer-v2.9.0-CVE-2022-23467-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v2.9.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v2.9.0-CVE-2022-23467-TP.c |
| Line | 104 | 104 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | openrazer@@openrazer-v2.9.0-CVE-2022-23467-TP.c |
| Method | int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max) |

```
....
104.        memcpy(response_report, buf, sizeof(struct razer_report));
```

**Dangerous Functions\Path 4:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=960 |
| Status | New |

The dangerous function, memcpy, was found in use at line 74 in openrazer@@openrazer-v3.0.0-CVE-2022-23467-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.0.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v3.0.0-CVE-2022-23467-TP.c |
| Line | 104 | 104 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | openrazer@@openrazer-v3.0.0-CVE-2022-23467-TP.c |
| Method | int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max) |

```
....
104.       memcpy(response_report, buf, sizeof(struct razer_report));
```

### Dangerous Functions\Path 5:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=961](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=961) |
| Status | New |

The dangerous function, memcpy, was found in use at line 74 in openrazer@@openrazer-v3.1.0-CVE-2022-23467-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.1.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v3.1.0-CVE-2022-23467-TP.c |
| Line | 104 | 104 |
| Object | memcpy | memcpy |

Code Snippet
File Name openrazer@@openrazer-v3.1.0-CVE-2022-23467-TP.c
Method int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max)

```
....
104.       memcpy(response_report, buf, sizeof(struct razer_report));
```

### Dangerous Functions\Path 6:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=962](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=962) |
| Status | New |

The dangerous function, memcpy, was found in use at line 74 in openrazer@@openrazer-v3.2.0-CVE-2022-23467-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.2.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v3.2.0-CVE-2022-23467-TP.c |
| Line | 104 | 104 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | openrazer@@openrazer-v3.2.0-CVE-2022-23467-TP.c |
| Method | int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max) |

```
....
104.       memcpy(response_report, buf, sizeof(struct razer_report));
```

**Dangerous Functions\Path 7:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=963 |
| Status | New |

The dangerous function, memcpy, was found in use at line 74 in openrazer@@openrazer-v3.3.0-CVE-2022-23467-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.3.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v3.3.0-CVE-2022-23467-TP.c |
| Line | 104 | 104 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | openrazer@@openrazer-v3.3.0-CVE-2022-23467-TP.c |
| Method | int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max) |

```
....
104.       memcpy(response_report, buf, sizeof(struct razer_report));
```

**Dangerous Functions\Path 8:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=964 |
| Status | New |

The dangerous function, memcpy, was found in use at line 71 in openrazer@@openrazer-v3.4.0-CVE-2022-23467-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.4.0-CVE- | openrazer@@openrazer-v3.4.0-CVE- |

| | | |
|---|---|---|
| | 2022-23467-TP.c | 2022-23467-TP.c |
| Line | 101 | 101 |
| Object | memcpy | memcpy |

Code Snippet
File Name openrazer@@openrazer-v3.4.0-CVE-2022-23467-TP.c
Method int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max)

```
....
101.        memcpy(response_report, buf, sizeof(struct razer_report));
```

**Dangerous Functions\Path 9:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=965](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=965) |
| Status | New |

The dangerous function, memcpy, was found in use at line 237 in openrazer@@openrazer-v3.5.0-CVE-2022-23467-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.5.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v3.5.0-CVE-2022-23467-TP.c |
| Line | 265 | 265 |
| Object | memcpy | memcpy |

Code Snippet
File Name openrazer@@openrazer-v3.5.0-CVE-2022-23467-TP.c
Method int razer_send_argb_msg(struct usb_device* usb_dev, unsigned char channel, unsigned char size, void const* data)

```
....
265.        memcpy(report.color_data, data, size * 3);
```

**Dangerous Functions\Path 10:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=966](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=966) |
| Status | New |

The dangerous function, memcpy, was found in use at line 71 in openrazer@@openrazer-v3.5.0-CVE-2022-23467-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.5.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v3.5.0-CVE-2022-23467-TP.c |
| Line | 101 | 101 |
| Object | memcpy | memcpy |

Code Snippet
File Name openrazer@@openrazer-v3.5.0-CVE-2022-23467-TP.c
Method int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max)

```
....
101.        memcpy(response_report, buf, sizeof(struct razer_report));
```

**Dangerous Functions\Path 11:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=967 |
| Status | New |

The dangerous function, memcpy, was found in use at line 245 in openrazer@@openrazer-v3.6.0-CVE-2022-23467-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.6.0-CVE-2022-23467-FP.c | openrazer@@openrazer-v3.6.0-CVE-2022-23467-FP.c |
| Line | 273 | 273 |
| Object | memcpy | memcpy |

Code Snippet
File Name openrazer@@openrazer-v3.6.0-CVE-2022-23467-FP.c
Method int razer_send_argb_msg(struct usb_device* usb_dev, unsigned char channel, unsigned char size, void const* data)

```
....
273.        memcpy(report.color_data, data, size * 3);
```

**Dangerous Functions\Path 12:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=968 |
| Status | New |

The dangerous function, memcpy, was found in use at line 71 in openrazer@@openrazer-v3.6.0-CVE-2022-23467-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.6.0-CVE-2022-23467-FP.c | openrazer@@openrazer-v3.6.0-CVE-2022-23467-FP.c |
| Line | 101 | 101 |
| Object | memcpy | memcpy |

Code Snippet

File Name openrazer@@openrazer-v3.6.0-CVE-2022-23467-FP.c

Method int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max)

```
....
101.       memcpy(response_report, buf, sizeof(struct razer_report));
```

**Dangerous Functions\Path 13:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=969 |
| Status | New |

The dangerous function, memcpy, was found in use at line 247 in openrazer@@openrazer-v3.7.0-CVE-2022-23467-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.7.0-CVE-2022-23467-FP.c | openrazer@@openrazer-v3.7.0-CVE-2022-23467-FP.c |
| Line | 275 | 275 |
| Object | memcpy | memcpy |

Code Snippet

File Name openrazer@@openrazer-v3.7.0-CVE-2022-23467-FP.c

Method int razer_send_argb_msg(struct usb_device* usb_dev, unsigned char channel, unsigned char size, void const* data)

```
....
275.       memcpy(report.color_data, data, size * 3);
```

### Dangerous Functions\Path 14:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=970](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=970) |
| Status | New |

The dangerous function, memcpy, was found in use at line 70 in openrazer@@openrazer-v3.7.0-CVE-2022-23467-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.7.0-CVE-2022-23467-FP.c | openrazer@@openrazer-v3.7.0-CVE-2022-23467-FP.c |
| Line | 104 | 104 |
| Object | memcpy | memcpy |

| Code Snippet | |
|---|---|
| File Name | openrazer@@openrazer-v3.7.0-CVE-2022-23467-FP.c |
| Method | int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max) |

```
....
104.        memcpy(response_report, buf, sizeof(struct razer_report));
```

### Dangerous Functions\Path 15:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=971](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=971) |
| Status | New |

The dangerous function, memcpy, was found in use at line 159 in OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Line | 215 | 215 |
| Object | memcpy | memcpy |

| Code Snippet | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Method | static int idprime_process_index(sc_card_t *card, idprime_private_data_t *priv, int length) |

```
....
215.                    memcpy(priv->tinfo_df, new_object.df,
sizeof(priv->tinfo_df));
```

**Dangerous Functions\Path 16:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=972 |
| Status | New |

The dangerous function, memcpy, was found in use at line 361 in OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Line | 369 | 369 |
| Object | memcpy | memcpy |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c
Method static int idprime_fill_prkey_info(list_t *list, idprime_object_t **entry, sc_pkcs15_prkey_info_t *prkey_info)

```
....
369.          memcpy(prkey_info->path.value, (*entry)->df,
sizeof((*entry)->df));
```

**Dangerous Functions\Path 17:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=973 |
| Status | New |

The dangerous function, memcpy, was found in use at line 385 in OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Line | 411 | 411 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Method | static int idprime_get_serial(sc_card_t* card, sc_serial_number_t* serial) |

```
....
411.        memcpy(serial->value, buf, serial->len);
```

**Dangerous Functions\Path 18:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=974 |
| Status | New |

The dangerous function, memcpy, was found in use at line 415 in OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Line | 433 | 433 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Method | static int idprime_get_token_name(sc_card_t* card, char** tname) |

```
....
433.        memcpy(tinfo_path.value, priv->tinfo_df, 2);
```

**Dangerous Functions\Path 19:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=975 |
| Status | New |

The dangerous function, memcpy, was found in use at line 529 in OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Line | 585 | 585 |

| | | |
|---|---|---|
| Object | memcpy | memcpy |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c
Method static int idprime_read_binary(sc_card_t *card, unsigned int offset,

```
....
585.                    memcpy(priv->cache_buf, buffer, r);
```

**Dangerous Functions\Path 20:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=976 |
| Status | New |

The dangerous function, memcpy, was found in use at line 529 in OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Line | 594 | 594 |
| Object | memcpy | memcpy |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c
Method static int idprime_read_binary(sc_card_t *card, unsigned int offset,

```
....
594.          memcpy(buf, priv->cache_buf + offset, size);
```

**Dangerous Functions\Path 21:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=977 |
| Status | New |

The dangerous function, memcpy, was found in use at line 663 in OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |

| | | |
|---|---|---|
| Line | 683 | 683 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Method | idprime_compute_signature(struct sc_card *card, |

```
....
683.        memcpy(p, data, datalen);
```

**Dangerous Functions\Path 22:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=978 |
| Status | New |

The dangerous function, memcpy, was found in use at line 736 in OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Line | 765 | 765 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Method | idprime_decipher(struct sc_card *card, |

```
....
765.        memcpy(sbuf + 1, crgram, crgram_len);
```

**Dangerous Functions\Path 23:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=979 |
| Status | New |

The dangerous function, memcpy, was found in use at line 465 in OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE- | OpenSC@@OpenSC-0.21.0-rc1-CVE- |

| | | |
|---|---|---|
| | 2023-2977-TP.c | 2023-2977-TP.c |
| Line | 489 | 489 |
| Object | memcpy | memcpy |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c
Method cardos_store_pin(sc_profile_t *profile, sc_card_t *card,

```
....
489.        memcpy(pinpadded, pin, pin_len);
```

**Dangerous Functions\Path 24:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=980 |
| Status | New |

The dangerous function, memcpy, was found in use at line 754 in OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c |
| Line | 777 | 777 |
| Object | memcpy | memcpy |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c
Method static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len,

```
....
777.        memcpy(pubkey->u.rsa.modulus.data, p, tlen);
```

**Dangerous Functions\Path 25:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=981 |
| Status | New |

The dangerous function, memcpy, was found in use at line 754 in OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|

| | | |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c |
| Line | 788 | 788 |
| Object | memcpy | memcpy |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c
Method static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len,

```
....
788.         memcpy(pubkey->u.rsa.exponent.data, p, tlen);
```

**Dangerous Functions\Path 26:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=982 |
| Status | New |

The dangerous function, memcpy, was found in use at line 794 in OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c |
| Line | 811 | 811 |
| Object | memcpy | memcpy |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c
Method do_cardos_extract_pubkey(sc_card_t *card, int nr, u8 tag,

```
....
811.         memcpy(bn->data, buf + 4, count);
```

**Dangerous Functions\Path 27:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=983 |
| Status | New |

The dangerous function, memcpy, was found in use at line 84 in OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Line | 205 | 205 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Method | sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj, |

```
....
205.        memcpy(obj->data, &info, sizeof(info));
```

**Dangerous Functions\Path 28:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=984 |
| Status | New |

The dangerous function, memcpy, was found in use at line 352 in OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Line | 434 | 434 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Method | int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card, |

```
....
434.                memcpy(&data.pin2, &data.pin1, sizeof (data.pin1));
```

**Dangerous Functions\Path 29:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=985 |
| Status | New |

The dangerous function, memcpy, was found in use at line 465 in OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c |
| Line | 489 | 489 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c |
| Method | cardos_store_pin(sc_profile_t *profile, sc_card_t *card, |

```
....
489.        memcpy(pinpadded, pin, pin_len);
```

**Dangerous Functions\Path 30:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=986 |
| Status | New |

The dangerous function, memcpy, was found in use at line 754 in OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c |
| Line | 777 | 777 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c |
| Method | static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len, |

```
....
777.        memcpy(pubkey->u.rsa.modulus.data, p, tlen);
```

**Dangerous Functions\Path 31:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=987 |
| Status | New |

The dangerous function, memcpy, was found in use at line 754 in OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c |
| Line | 788 | 788 |
| Object | memcpy | memcpy |

Code Snippet
File Name    OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c
Method    static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len,

```
....
788.        memcpy(pubkey->u.rsa.exponent.data, p, tlen);
```

**Dangerous Functions\Path 32:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=988 |
| Status | New |

The dangerous function, memcpy, was found in use at line 794 in OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c |
| Line | 811 | 811 |
| Object | memcpy | memcpy |

Code Snippet
File Name    OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c
Method    do_cardos_extract_pubkey(sc_card_t *card, int nr, u8 tag,

```
....
811.        memcpy(bn->data, buf + 4, count);
```

**Dangerous Functions\Path 33:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=989 |

| | |
|---|---|
| Status | New |

The dangerous function, memcpy, was found in use at line 84 in OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Line | 205 | 205 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Method | sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj, |

```
....
205.        memcpy(obj->data, &info, sizeof(info));
```

**Dangerous Functions\Path 34:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=990 |
| Status | New |

The dangerous function, memcpy, was found in use at line 352 in OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Line | 434 | 434 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Method | int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card, |

```
....
434.                memcpy(&data.pin2, &data.pin1, sizeof (data.pin1));
```

**Dangerous Functions\Path 35:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=991 |
| Status | New |

The dangerous function, memcpy, was found in use at line 465 in OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c |
| Line | 489 | 489 |
| Object | memcpy | memcpy |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c
Method cardos_store_pin(sc_profile_t *profile, sc_card_t *card,

```
....
489.        memcpy(pinpadded, pin, pin_len);
```

**Dangerous Functions\Path 36:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=992 |
| Status | New |

The dangerous function, memcpy, was found in use at line 754 in OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c |
| Line | 777 | 777 |
| Object | memcpy | memcpy |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c
Method static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len,

```
....
777.        memcpy(pubkey->u.rsa.modulus.data, p, tlen);
```

**Dangerous Functions\Path 37:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=993 |
| Status | New |

The dangerous function, memcpy, was found in use at line 754 in OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c |
| Line | 788 | 788 |
| Object | memcpy | memcpy |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c
Method static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len,

```
....
788.        memcpy(pubkey->u.rsa.exponent.data, p, tlen);
```

**Dangerous Functions\Path 38:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=994 |
| Status | New |

The dangerous function, memcpy, was found in use at line 794 in OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c |
| Line | 811 | 811 |
| Object | memcpy | memcpy |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c
Method do_cardos_extract_pubkey(sc_card_t *card, int nr, u8 tag,

```
....
811.        memcpy(bn->data, buf + 4, count);
```

**Dangerous Functions\Path 39:**

| | |
|---|---|
| Severity | Medium |

| | |
|---|---|
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=995](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=995) |
| Status | New |

The dangerous function, memcpy, was found in use at line 84 in OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c |
| Line | 205 | 205 |
| Object | memcpy | memcpy |

Code Snippet

File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c

Method sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj,

```
....
205.        memcpy(obj->data, &info, sizeof(info));
```

**Dangerous Functions\Path 40:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=996](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=996) |
| Status | New |

The dangerous function, memcpy, was found in use at line 352 in OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c |
| Line | 434 | 434 |
| Object | memcpy | memcpy |

Code Snippet

File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c

Method int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card,

```
....
434.                memcpy(&data.pin2, &data.pin1, sizeof (data.pin1));
```

**Dangerous Functions\Path 41:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=997 |
| Status | New |

The dangerous function, memcpy, was found in use at line 465 in OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c |
| Line | 489 | 489 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c |
| Method | cardos_store_pin(sc_profile_t *profile, sc_card_t *card, |

```
....
489.        memcpy(pinpadded, pin, pin_len);
```

**Dangerous Functions\Path 42:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=998 |
| Status | New |

The dangerous function, memcpy, was found in use at line 754 in OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c |
| Line | 777 | 777 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c |
| Method | static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len, |

```
....
777.        memcpy(pubkey->u.rsa.modulus.data, p, tlen);
```

**Dangerous Functions\Path 43:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=999 |
| Status | New |

The dangerous function, memcpy, was found in use at line 754 in OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c |
| Line | 788 | 788 |
| Object | memcpy | memcpy |

Code Snippet
File Name OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c
Method static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len,

```
....
788.        memcpy(pubkey->u.rsa.exponent.data, p, tlen);
```

**Dangerous Functions\Path 44:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=1000 |
| Status | New |

The dangerous function, memcpy, was found in use at line 794 in OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c |
| Line | 811 | 811 |
| Object | memcpy | memcpy |

Code Snippet
File Name OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c
Method do_cardos_extract_pubkey(sc_card_t *card, int nr, u8 tag,

```
....
811.        memcpy(bn->data, buf + 4, count);
```

**Dangerous Functions\Path 45:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=1001](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=1001) |
| Status | New |

The dangerous function, memcpy, was found in use at line 84 in OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c |
| Line | 205 | 205 |
| Object | memcpy | memcpy |

Code Snippet
File Name OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c
Method sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj,

```
....
205.        memcpy(obj->data, &info, sizeof(info));
```

**Dangerous Functions\Path 46:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=1002](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=1002) |
| Status | New |

The dangerous function, memcpy, was found in use at line 352 in OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c |
| Line | 434 | 434 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c |
| Method | int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card, |

```
....
434.                memcpy(&data.pin2, &data.pin1, sizeof (data.pin1));
```

**Dangerous Functions\Path 47:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=1003 |
| Status | New |

The dangerous function, memcpy, was found in use at line 143 in OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c |
| Line | 167 | 167 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c |
| Method | myeid_select_aid(struct sc_card *card, struct sc_aid *aid, unsigned char *out, size_t *out_len) |

```
....
167.                memcpy(out, apdu.resp, apdu.resplen);
```

**Dangerous Functions\Path 48:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=1004 |
| Status | New |

The dangerous function, memcpy, was found in use at line 516 in OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c |
| Line | 605 | 605 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c |
| Method | static int encode_file_structure(sc_card_t *card, const sc_file_t *file, |

```
....
605.            memcpy(&buf[20], file->prop_attr, 2);
```

**Dangerous Functions\Path 49:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=1005 |
| Status | New |

The dangerous function, memcpy, was found in use at line 725 in OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c |
| Line | 791 | 791 |
| Object | memcpy | memcpy |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c |
| Method | static int myeid_set_security_env_rsa(sc_card_t *card, const sc_security_env_t *env, |

```
....
791.              memcpy(p, env->file_ref.value, 2);
```

**Dangerous Functions\Path 50:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=1006 |
| Status | New |

The dangerous function, memcpy, was found in use at line 725 in OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c |
| Line | 817 | 817 |

|  |  |  |
|---|---|---|
| Object | memcpy | memcpy |

Code Snippet
File Name OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c
Method static int myeid_set_security_env_rsa(sc_card_t *card, const sc_security_env_t *env,

```
....
817.                    memcpy(p, target_file->value, 2);
```

## Buffer Overflow boundcpy WrongSizeParam

Query Path:
CPP\Cx\CPP Buffer Overflow\Buffer Overflow boundcpy WrongSizeParam Version:1

### Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.2 - Buffer overflows
OWASP Top 10 2017: A1-Injection

*Description*

**Buffer Overflow boundcpy WrongSizeParam\Path 1:**

|  |  |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=91 |
| Status | New |

The size of the buffer used by razer_get_usb_response in razer_report, at line 74 of openrazer@@openrazer-v2.7.0-CVE-2022-23467-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that razer_get_usb_response passes to razer_report, at line 74 of openrazer@@openrazer-v2.7.0-CVE-2022-23467-TP.c, to overwrite the target buffer.

|  | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v2.7.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v2.7.0-CVE-2022-23467-TP.c |
| Line | 104 | 104 |
| Object | razer_report | razer_report |

Code Snippet
File Name openrazer@@openrazer-v2.7.0-CVE-2022-23467-TP.c
Method int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max)

```
....
104.      memcpy(response_report, buf, sizeof(struct razer_report));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 2:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=92 |
| Status | New |

The size of the buffer used by razer_get_usb_response in razer_report, at line 74 of openrazer@@openrazer-v2.8.0-CVE-2022-23467-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that razer_get_usb_response passes to razer_report, at line 74 of openrazer@@openrazer-v2.8.0-CVE-2022-23467-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v2.8.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v2.8.0-CVE-2022-23467-TP.c |
| Line | 104 | 104 |
| Object | razer_report | razer_report |

Code Snippet

| | |
|---|---|
| File Name | openrazer@@openrazer-v2.8.0-CVE-2022-23467-TP.c |
| Method | int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max) |

```
....
104.      memcpy(response_report, buf, sizeof(struct razer_report));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 3:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=93 |
| Status | New |

The size of the buffer used by razer_get_usb_response in razer_report, at line 74 of openrazer@@openrazer-v2.9.0-CVE-2022-23467-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that razer_get_usb_response passes to razer_report, at line 74 of openrazer@@openrazer-v2.9.0-CVE-2022-23467-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v2.9.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v2.9.0-CVE-2022-23467-TP.c |
| Line | 104 | 104 |
| Object | razer_report | razer_report |

Code Snippet

| | |
|---|---|
| File Name | openrazer@@openrazer-v2.9.0-CVE-2022-23467-TP.c |
| Method | int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max) |

```
....
104.      memcpy(response_report, buf, sizeof(struct razer_report));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 4:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=94 |
| Status | New |

The size of the buffer used by razer_get_usb_response in razer_report, at line 74 of openrazer@@openrazer-v3.0.0-CVE-2022-23467-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that razer_get_usb_response passes to razer_report, at line 74 of openrazer@@openrazer-v3.0.0-CVE-2022-23467-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.0.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v3.0.0-CVE-2022-23467-TP.c |
| Line | 104 | 104 |
| Object | razer_report | razer_report |

Code Snippet

File Name openrazer@@openrazer-v3.0.0-CVE-2022-23467-TP.c

Method int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max)

```
....
104.        memcpy(response_report, buf, sizeof(struct razer_report));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 5:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=95 |
| Status | New |

The size of the buffer used by razer_get_usb_response in razer_report, at line 74 of openrazer@@openrazer-v3.1.0-CVE-2022-23467-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that razer_get_usb_response passes to razer_report, at line 74 of openrazer@@openrazer-v3.1.0-CVE-2022-23467-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.1.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v3.1.0-CVE-2022-23467-TP.c |
| Line | 104 | 104 |
| Object | razer_report | razer_report |

Code Snippet

File Name openrazer@@openrazer-v3.1.0-CVE-2022-23467-TP.c

Method int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max)

```
....
104.      memcpy(response_report, buf, sizeof(struct razer_report));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 6:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=96 |
| Status | New |

The size of the buffer used by razer_get_usb_response in razer_report, at line 74 of openrazer@@openrazer-v3.2.0-CVE-2022-23467-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that razer_get_usb_response passes to razer_report, at line 74 of openrazer@@openrazer-v3.2.0-CVE-2022-23467-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.2.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v3.2.0-CVE-2022-23467-TP.c |
| Line | 104 | 104 |
| Object | razer_report | razer_report |

Code Snippet
File Name openrazer@@openrazer-v3.2.0-CVE-2022-23467-TP.c
Method int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max)

```
....
104.      memcpy(response_report, buf, sizeof(struct razer_report));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 7:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=97 |
| Status | New |

The size of the buffer used by razer_get_usb_response in razer_report, at line 74 of openrazer@@openrazer-v3.3.0-CVE-2022-23467-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that razer_get_usb_response passes to razer_report, at line 74 of openrazer@@openrazer-v3.3.0-CVE-2022-23467-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.3.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v3.3.0-CVE-2022-23467-TP.c |

| | | |
|---|---|---|
| Line | 104 | 104 |
| Object | razer_report | razer_report |

Code Snippet
File Name openrazer@@openrazer-v3.3.0-CVE-2022-23467-TP.c
Method int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max)

```
....
104.       memcpy(response_report, buf, sizeof(struct razer_report));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 8:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=98 |
| Status | New |

The size of the buffer used by razer_get_usb_response in razer_report, at line 71 of openrazer@@openrazer-v3.4.0-CVE-2022-23467-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that razer_get_usb_response passes to razer_report, at line 71 of openrazer@@openrazer-v3.4.0-CVE-2022-23467-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.4.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v3.4.0-CVE-2022-23467-TP.c |
| Line | 101 | 101 |
| Object | razer_report | razer_report |

Code Snippet
File Name openrazer@@openrazer-v3.4.0-CVE-2022-23467-TP.c
Method int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max)

```
....
101.       memcpy(response_report, buf, sizeof(struct razer_report));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 9:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=99 |
| Status | New |

The size of the buffer used by razer_get_usb_response in razer_report, at line 71 of openrazer@@openrazer-v3.5.0-CVE-2022-23467-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that razer_get_usb_response passes to razer_report, at line 71 of openrazer@@openrazer-v3.5.0-CVE-2022-23467-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.5.0-CVE-2022-23467-TP.c | openrazer@@openrazer-v3.5.0-CVE-2022-23467-TP.c |
| Line | 101 | 101 |
| Object | razer_report | razer_report |

Code Snippet
File Name openrazer@@openrazer-v3.5.0-CVE-2022-23467-TP.c
Method int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max)

```
....
101.        memcpy(response_report, buf, sizeof(struct razer_report));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 10:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=100
Status New

The size of the buffer used by razer_get_usb_response in razer_report, at line 71 of openrazer@@openrazer-v3.6.0-CVE-2022-23467-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that razer_get_usb_response passes to razer_report, at line 71 of openrazer@@openrazer-v3.6.0-CVE-2022-23467-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.6.0-CVE-2022-23467-FP.c | openrazer@@openrazer-v3.6.0-CVE-2022-23467-FP.c |
| Line | 101 | 101 |
| Object | razer_report | razer_report |

Code Snippet
File Name openrazer@@openrazer-v3.6.0-CVE-2022-23467-FP.c
Method int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max)

```
....
101.        memcpy(response_report, buf, sizeof(struct razer_report));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 11:

Severity Medium
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20

| | |
|---|---|
| | 043&pathid=101 |
| Status | New |

The size of the buffer used by razer_get_usb_response in razer_report, at line 70 of openrazer@@openrazer-v3.7.0-CVE-2022-23467-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that razer_get_usb_response passes to razer_report, at line 70 of openrazer@@openrazer-v3.7.0-CVE-2022-23467-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | openrazer@@openrazer-v3.7.0-CVE-2022-23467-FP.c | openrazer@@openrazer-v3.7.0-CVE-2022-23467-FP.c |
| Line | 104 | 104 |
| Object | razer_report | razer_report |

Code Snippet

| | |
|---|---|
| File Name | openrazer@@openrazer-v3.7.0-CVE-2022-23467-FP.c |
| Method | int razer_get_usb_response(struct usb_device *usb_dev, uint report_index, struct razer_report* request_report, uint response_index, struct razer_report* response_report, ulong wait_min, ulong wait_max) |

```
....
104.        memcpy(response_report, buf, sizeof(struct razer_report));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 12:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=102 |
| Status | New |

The size of the buffer used by idprime_process_index in ->, at line 159 of OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that idprime_process_index passes to ->, at line 159 of OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Line | 215 | 215 |
| Object | -> | -> |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Method | static int idprime_process_index(sc_card_t *card, idprime_private_data_t *priv, int length) |

```
....
215.                        memcpy(priv->tinfo_df, new_object.df,
sizeof(priv->tinfo_df));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 13:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=103 |
| Status | New |

The size of the buffer used by idprime_fill_prkey_info in entry, at line 361 of OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that idprime_fill_prkey_info passes to entry, at line 361 of OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Line | 369 | 369 |
| Object | entry | entry |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Method | static int idprime_fill_prkey_info(list_t *list, idprime_object_t **entry, sc_pkcs15_prkey_info_t *prkey_info) |

```
....
369.        memcpy(prkey_info->path.value, (*entry)->df,
sizeof((*entry)->df));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 14:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=104 |
| Status | New |

The size of the buffer used by sc_pkcs15_decode_aodf_entry in info, at line 84 of OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that sc_pkcs15_decode_aodf_entry passes to info, at line 84 of OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Line | 205 | 205 |
| Object | info | info |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Method | sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj, |

```
....
205.         memcpy(obj->data, &info, sizeof(info));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 15:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=105 |
| Status | New |

The size of the buffer used by sc_pkcs15_verify_pin_with_session_pin in Namespace1402442233, at line 352 of OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that sc_pkcs15_verify_pin_with_session_pin passes to Namespace1402442233, at line 352 of OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Line | 434 | 434 |
| Object | Namespace1402442233 | Namespace1402442233 |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Method | int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card, |

```
....
434.                memcpy(&data.pin2, &data.pin1, sizeof (data.pin1));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 16:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=106 |
| Status | New |

The size of the buffer used by sc_pkcs15_decode_aodf_entry in info, at line 84 of OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that sc_pkcs15_decode_aodf_entry passes to info, at line 84 of OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Line | 205 | 205 |
| Object | info | info |

Code Snippet

|  |  |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Method | sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj, |

```
....
205.        memcpy(obj->data, &info, sizeof(info));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 17:

|  |  |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=107 |
| Status | New |

The size of the buffer used by sc_pkcs15_verify_pin_with_session_pin in Namespace1428709179, at line 352 of OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that sc_pkcs15_verify_pin_with_session_pin passes to Namespace1428709179, at line 352 of OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c, to overwrite the target buffer.

|  | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Line | 434 | 434 |
| Object | Namespace1428709179 | Namespace1428709179 |

Code Snippet

|  |  |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Method | int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card, |

```
....
434.            memcpy(&data.pin2, &data.pin1, sizeof (data.pin1));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 18:

|  |  |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=108 |
| Status | New |

The size of the buffer used by sc_pkcs15_decode_aodf_entry in info, at line 84 of OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that sc_pkcs15_decode_aodf_entry passes to info, at line 84 of OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c, to overwrite the target buffer.

|  | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c |
| Line | 205 | 205 |

| | | |
|---|---|---|
| Object | info | info |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c
Method sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj,

```
....
205.        memcpy(obj->data, &info, sizeof(info));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 19:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=109](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=109) |
| Status | New |

The size of the buffer used by sc_pkcs15_verify_pin_with_session_pin in Namespace1817123562, at line 352 of OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that sc_pkcs15_verify_pin_with_session_pin passes to Namespace1817123562, at line 352 of OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c |
| Line | 434 | 434 |
| Object | Namespace1817123562 | Namespace1817123562 |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c
Method int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card,

```
....
434.            memcpy(&data.pin2, &data.pin1, sizeof (data.pin1));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 20:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=110](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=110) |
| Status | New |

The size of the buffer used by sc_pkcs15_decode_aodf_entry in info, at line 84 of OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that sc_pkcs15_decode_aodf_entry passes to info, at line 84 of OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|

| | | |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c |
| Line | 205 | 205 |
| Object | info | info |

Code Snippet
File Name OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c
Method sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj,

```
....
205.        memcpy(obj->data, &info, sizeof(info));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 21:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=111 |
| Status | New |

The size of the buffer used by sc_pkcs15_verify_pin_with_session_pin in Namespace355955203, at line 352 of OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that sc_pkcs15_verify_pin_with_session_pin passes to Namespace355955203, at line 352 of OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c |
| Line | 434 | 434 |
| Object | Namespace355955203 | Namespace355955203 |

Code Snippet
File Name OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c
Method int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card,

```
....
434.              memcpy(&data.pin2, &data.pin1, sizeof (data.pin1));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 22:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=112 |
| Status | New |

The size of the buffer used by myeid_compute_raw_2048_signature in sc_security_env_t, at line 1110 of OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that myeid_compute_raw_2048_signature passes to sc_security_env_t, at line 1110 of OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c |
| Line | 1127 | 1127 |
| Object | sc_security_env_t | sc_security_env_t |

Code Snippet
File Name OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c
Method myeid_compute_raw_2048_signature(struct sc_card *card, const u8 * data, size_t datalen,

```
....
1127.        memcpy(&env, priv->sec_env, sizeof(sc_security_env_t));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 23:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=113 |
| Status | New |

The size of the buffer used by sc_pkcs15_decode_aodf_entry in info, at line 84 of OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that sc_pkcs15_decode_aodf_entry passes to info, at line 84 of OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c |
| Line | 205 | 205 |
| Object | info | info |

Code Snippet
File Name OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c
Method sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj,

```
....
205.        memcpy(obj->data, &info, sizeof(info));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 24:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=114 |

| | |
|---|---|
| Status | New |

The size of the buffer used by sc_pkcs15_verify_pin_with_session_pin in Namespace701199150, at line 339 of OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that sc_pkcs15_verify_pin_with_session_pin passes to Namespace701199150, at line 339 of OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c |
| Line | 421 | 421 |
| Object | Namespace701199150 | Namespace701199150 |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c |
| Method | int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card, |

```
....
421.            memcpy(&data.pin2, &data.pin1, sizeof (data.pin1));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 25:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=115 |
| Status | New |

The size of the buffer used by sc_pkcs15_decode_aodf_entry in info, at line 84 of OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that sc_pkcs15_decode_aodf_entry passes to info, at line 84 of OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c |
| Line | 205 | 205 |
| Object | info | info |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c |
| Method | sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj, |

```
....
205.        memcpy(obj->data, &info, sizeof(info));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 26:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=116 |
| Status | New |

The size of the buffer used by sc_pkcs15_verify_pin_with_session_pin in Namespace467264499, at line 340 of OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that sc_pkcs15_verify_pin_with_session_pin passes to Namespace467264499, at line 340 of OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c |
| Line | 422 | 422 |
| Object | Namespace467264499 | Namespace467264499 |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c |
| Method | int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card, |

```
....
422.            memcpy(&data.pin2, &data.pin1, sizeof (data.pin1));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 27:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=117 |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c |
| Line | 124 | 124 |
| Object | str | str |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c |
| Method | static inline char* parse_to_param(char *buffer, char *end, |

```
....
124.                                add_param( param , to_b );
```

**Buffer Overflow boundcpy WrongSizeParam\Path 28:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=118 |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c |
| Line | 157 | 157 |
| Object | str | str |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c |
| Method | static inline char* parse_to_param(char *buffer, char *end, |

```
....
157.                                                add_param( param , to_b );
```

**Buffer Overflow boundcpy WrongSizeParam\Path 29:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=119 |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c |
| Line | 193 | 193 |
| Object | str | str |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c |
| Method | static inline char* parse_to_param(char *buffer, char *end, |

```
....
193.                                                add_param( param , to_b );
```

### Buffer Overflow boundcpy WrongSizeParam\Path 30:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=120 |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c |
| Line | 220 | 220 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c
Method static inline char* parse_to_param(char *buffer, char *end,

```
....
220.                                    add_param( param , to_b );
```

### Buffer Overflow boundcpy WrongSizeParam\Path 31:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=121 |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c |
| Line | 258 | 258 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c
Method static inline char* parse_to_param(char *buffer, char *end,

```
....
258.                                    add_param( param , to_b );
```

### Buffer Overflow boundcpy WrongSizeParam\Path 32:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=122](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=122) |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c |
| Line | 284 | 284 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c
Method static inline char* parse_to_param(char *buffer, char *end,

```
....
284.                                    add_param(param,to_b);
```

### Buffer Overflow boundcpy WrongSizeParam\Path 33:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=123](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=123) |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c |
| Line | 465 | 465 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27599-TP.c
Method static inline char* parse_to_param(char *buffer, char *end,

```
....
465.                        add_param(param, to_b);
```

### Buffer Overflow boundcpy WrongSizeParam\Path 34:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=124 |
| Status | New |

The size of the buffer used by *create_reference in cJSON, at line 1775 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *create_reference passes to cJSON, at line 1775 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 1782 | 1782 |
| Object | cJSON | cJSON |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Method | static cJSON *create_reference(const cJSON *item) |

```
....
1782.       memcpy(ref, item, sizeof(cJSON));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 35:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=125 |
| Status | New |

The size of the buffer used by reindex_dests in ds_dest_t, at line 442 of OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that reindex_dests passes to ds_dest_t, at line 442 of OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c |
| Line | 466 | 466 |
| Object | ds_dest_t | ds_dest_t |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c |

Method int reindex_dests( ds_data_t *d_data)

```
....
466.                        memcpy(&dp0[j], sp->dlist, sizeof(ds_dest_t));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 36:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=126](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=126) |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c |
| Line | 124 | 124 |
| Object | str | str |

Code Snippet

File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c

Method static inline char* parse_to_param(char *buffer, char *end,

```
....
124.                                        add_param( param , to_b );
```

**Buffer Overflow boundcpy WrongSizeParam\Path 37:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=127](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=127) |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c |
| Line | 157 | 157 |
| Object | str | str |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c |
| Method | static inline char* parse_to_param(char *buffer, char *end, |

```
....
157.                                        add_param( param , to_b );
```

### Buffer Overflow boundcpy WrongSizeParam\Path 38:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=128 |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c |
| Line | 193 | 193 |
| Object | str | str |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c |
| Method | static inline char* parse_to_param(char *buffer, char *end, |

```
....
193.                                        add_param( param , to_b );
```

### Buffer Overflow boundcpy WrongSizeParam\Path 39:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=129 |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c |
| Line | 220 | 220 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c
Method static inline char* parse_to_param(char *buffer, char *end,

```
....
220.                                        add_param( param , to_b );
```

### Buffer Overflow boundcpy WrongSizeParam\Path 40:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=130 |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c |
| Line | 258 | 258 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c
Method static inline char* parse_to_param(char *buffer, char *end,

```
....
258.                                        add_param( param , to_b );
```

### Buffer Overflow boundcpy WrongSizeParam\Path 41:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=131 |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c |
| Line | 284 | 284 |
| Object | str | str |

| | |
|---|---|
| Code Snippet | |
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c |
| Method | static inline char* parse_to_param(char *buffer, char *end, |

```
....
284.                                             add_param(param,to_b);
```

### Buffer Overflow boundcpy WrongSizeParam\Path 42:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=132 |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c |
| Line | 465 | 465 |
| Object | str | str |

| | |
|---|---|
| Code Snippet | |
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27599-FP.c |
| Method | static inline char* parse_to_param(char *buffer, char *end, |

```
....
465.                          add_param(param, to_b);
```

### Buffer Overflow boundcpy WrongSizeParam\Path 43:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=133 |
| Status | New |

The size of the buffer used by *create_reference in cJSON, at line 1824 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *create_reference passes to cJSON, at line 1824 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 1831 | 1831 |

| | | |
|---|---|---|
| Object | cJSON | cJSON |

Code Snippet

File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c

Method static cJSON *create_reference(const cJSON *item)

```
....
1831.       memcpy(ref, item, sizeof(cJSON));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 44:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=134 |
| Status | New |

The size of the buffer used by reindex_dests in ds_dest_t, at line 426 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that reindex_dests passes to ds_dest_t, at line 426 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c |
| Line | 450 | 450 |
| Object | ds_dest_t | ds_dest_t |

Code Snippet

File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c

Method int reindex_dests( ds_data_t *d_data)

```
....
450.                    memcpy(&dp0[j], sp->dlist, sizeof(ds_dest_t));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 45:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=135 |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c |

| | | |
|---|---|---|
| Line | 124 | 124 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c
Method static inline char* parse_to_param(char *buffer, char *end,

```
....
124.                                    add_param( param , to_b );
```

**Buffer Overflow boundcpy WrongSizeParam\Path 46:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=136 |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c |
| Line | 157 | 157 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c
Method static inline char* parse_to_param(char *buffer, char *end,

```
....
157.                                    add_param( param , to_b );
```

**Buffer Overflow boundcpy WrongSizeParam\Path 47:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=137 |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023- | OpenSIPS@@opensips-3.1.1-CVE-2023- |

| | | |
|---|---|---|
| | 27599-TP.c | 27599-TP.c |
| Line | 193 | 193 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c
Method static inline char* parse_to_param(char *buffer, char *end,

```
....
193.                                            add_param( param , to_b );
```

**Buffer Overflow boundcpy WrongSizeParam\Path 48:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=138 |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c |
| Line | 220 | 220 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c
Method static inline char* parse_to_param(char *buffer, char *end,

```
....
220.                                            add_param( param , to_b );
```

**Buffer Overflow boundcpy WrongSizeParam\Path 49:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=139 |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|

| | | |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c |
| Line | 258 | 258 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c
Method static inline char* parse_to_param(char *buffer, char *end,

```
....
258.                                            add_param( param , to_b );
```

**Buffer Overflow boundcpy WrongSizeParam\Path 50:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=140](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=140) |
| Status | New |

The size of the buffer used by parse_to_param in str, at line 92 of OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_to_param passes to str, at line 92 of OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c |
| Line | 284 | 284 |
| Object | str | str |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27599-TP.c
Method static inline char* parse_to_param(char *buffer, char *end,

```
....
284.                                            add_param(param,to_b);
```

## Use of Zero Initialized Pointer

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

### Categories

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

*Description*

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

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2645 |
| Status | New |

The variable declared in tp at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 2033 is not initialized when it is used by tp at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 2033.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 2038 | 2062 |
| Object | tp | tp |

Code Snippet

File Name openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c

Method ngx_http_lua_ffi_shdict_set_expire(ngx_shm_zone_t *zone, u_char *key,

```
....
2038.      ngx_time_t                  *tp = NULL;
....
2062.          sd->expires = (uint64_t) tp->sec * 1000 + tp->msec
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2646 |
| Status | New |

The variable declared in tp at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 2033 is not initialized when it is used by tp at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 2033.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 2038 | 2062 |
| Object | tp | tp |

Code Snippet

File Name openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c

Method ngx_http_lua_ffi_shdict_set_expire(ngx_shm_zone_t *zone, u_char *key,

```
....
2038.      ngx_time_t                  *tp = NULL;
....
2062.          sd->expires = (uint64_t) tp->sec * 1000 + tp->msec
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2647 |
| Status | New |

The variable declared in tp at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 2033 is not initialized when it is used by tp at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 2033.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 2038 | 2062 |
| Object | tp | tp |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_set_expire(ngx_shm_zone_t *zone, u_char *key,

```
....
2038.       ngx_time_t                       *tp = NULL;
....
2062.           sd->expires = (uint64_t) tp->sec * 1000 + tp->msec
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2648 |
| Status | New |

The variable declared in tp at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 2033 is not initialized when it is used by tp at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 2033.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 2038 | 2062 |
| Object | tp | tp |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_set_expire(ngx_shm_zone_t *zone, u_char *key,

```
....
2038.      ngx_time_t                   *tp = NULL;
....
2062.          sd->expires = (uint64_t) tp->sec * 1000 + tp->msec
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2649 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c in line 133 is not initialized when it is used by file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c in line 133.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Line | 136 | 149 |
| Object | file | file |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Method | static int idprime_select_index(sc_card_t *card) |

```
....
136.        sc_file_t *file = NULL;
....
149.              r = file->size;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2650 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c in line 385 is not initialized when it is used by file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c in line 385.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Line | 388 | 402 |
| Object | file | file |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Method | static int idprime_get_serial(sc_card_t* card, sc_serial_number_t* serial) |

```
....
388.        sc_file_t *file = NULL;
....
402.        r = iso_ops->read_binary(card, 0, buf, file->size, 0);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2651 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c in line 409 is not initialized when it is used by file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c in line 409.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c |
| Line | 414 | 448 |
| Object | file | file |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c |
| Method | cardos_delete_object(sc_profile_t *profile, struct sc_pkcs15_card *p15card, |

```
....
414.        sc_file_t *file = NULL;
....
448.              stored_in_ef = (file->type != SC_FILE_TYPE_DF);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2652 |
| Status | New |

The variable declared in skey_obj at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 352 is not initialized when it is used by skey_obj at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 352.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Line | 405 | 416 |
| Object | skey_obj | skey_obj |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c
Method int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card,

```
....
405.            struct sc_pkcs15_object *skey_obj = NULL;
....
416.            sc_log(ctx, "found secret key '%s'", skey_obj->label);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2653 |
| Status | New |

The variable declared in skey_obj at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 352 is not initialized when it is used by skey_obj at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 352.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Line | 405 | 414 |
| Object | skey_obj | skey_obj |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c
Method int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card,

```
....
405.            struct sc_pkcs15_object *skey_obj = NULL;
....
414.            skey_info = (struct sc_pkcs15_skey_info *)skey_obj-
>data;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2654 |
| Status | New |

The variable declared in puk_info at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 575 is not initialized when it is used by puk_info at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 575.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |

| | | |
|---|---|---|
| Line | 583 | 647 |
| Object | puk_info | puk_info |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c
Method int sc_pkcs15_unblock_pin(struct sc_pkcs15_card *p15card,

```
....
583.        struct sc_pkcs15_auth_info *puk_info = NULL;
....
647.        data.pin1.pad_length = puk_info->attrs.pin.stored_length;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2655 |
| Status | New |

The variable declared in puk_info at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 575 is not initialized when it is used by puk_info at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 575.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Line | 583 | 646 |
| Object | puk_info | puk_info |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c
Method int sc_pkcs15_unblock_pin(struct sc_pkcs15_card *p15card,

```
....
583.        struct sc_pkcs15_auth_info *puk_info = NULL;
....
646.        data.pin1.max_length = puk_info->attrs.pin.max_length;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2656 |
| Status | New |

The variable declared in puk_info at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 575 is not initialized when it is used by puk_info at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 575.

| | Source | Destination |
|---|---|---|

| | | |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Line | 583 | 645 |
| Object | puk_info | puk_info |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c
Method int sc_pkcs15_unblock_pin(struct sc_pkcs15_card *p15card,

```
....
583.        struct sc_pkcs15_auth_info *puk_info = NULL;
....
645.        data.pin1.min_length = puk_info->attrs.pin.min_length;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2657 |
| Status | New |

The variable declared in puk_info at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 575 is not initialized when it is used by puk_info at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 575.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Line | 583 | 644 |
| Object | puk_info | puk_info |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c
Method int sc_pkcs15_unblock_pin(struct sc_pkcs15_card *p15card,

```
....
583.        struct sc_pkcs15_auth_info *puk_info = NULL;
....
644.        data.pin1.pad_char    = puk_info->attrs.pin.pad_char;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2658 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 147 is not initialized when it is used by file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 147.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 150 | 191 |
| Object | file | file |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_erase_card(struct sc_profile *profile, struct sc_pkcs15_card *p15card)

```
....
150.        struct sc_file  *file = NULL;
....
191.              rv = sc_erase_binary(p15card->card, 0, file->size, 0);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2659 |
| Status | New |

The variable declared in file_p_prvkey at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 521 is not initialized when it is used by acl at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 412.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 528 | 422 |
| Object | file_p_prvkey | acl |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_create_key(struct sc_profile *profile, struct sc_pkcs15_card *p15card,

```
....
528.        struct sc_file    *file_p_prvkey = NULL, *parent = NULL;
```

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_fix_file_access_rule(struct sc_pkcs15_card *p15card, struct sc_file *file,

```
....
422.        acl = sc_file_get_acl_entry(file, ac_op);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2660 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by acl at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 412.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 422 |
| Object | file | acl |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.         struct sc_file     *file = NULL;
```

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_fix_file_access_rule(struct sc_pkcs15_card *p15card, struct sc_file *file,

```
....
422.         acl = sc_file_get_acl_entry(file, ac_op);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2661 |
| Status | New |

The variable declared in file_p_prvkey at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 521 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 521.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 528 | 589 |
| Object | file_p_prvkey | sdo |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_create_key(struct sc_profile *profile, struct sc_pkcs15_card *p15card, |

```
....
528.         struct sc_file    *file_p_prvkey = NULL, *parent = NULL;
....
589.         sdo->file = file_p_prvkey;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2662 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 521.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 589 |
| Object | file | sdo |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card, |

```
....
217.         struct sc_file    *file = NULL;
```

▼

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_create_key(struct sc_profile *profile, struct sc_pkcs15_card *p15card, |

```
....
589.         sdo->file = file_p_prvkey;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2663 |
| Status | New |

The variable declared in sdo at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 521 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 521.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 526 | 592 |
| Object | sdo | sdo |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_create_key(struct sc_profile *profile, struct sc_pkcs15_card *p15card,

```
....
526.        struct sc_authentic_sdo *sdo = NULL;
....
592.        rv = sc_pkcs15_allocate_object_content(ctx, object, 
(unsigned char *)sdo, sizeof(struct sc_authentic_sdo));
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2664 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 334 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 521.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 339 | 592 |
| Object | file | sdo |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
339.        struct sc_file *file = NULL;
```

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_create_key(struct sc_profile *profile, struct sc_pkcs15_card *p15card,

```
....
592.        rv = sc_pkcs15_allocate_object_content(ctx, object,
(unsigned char *)sdo, sizeof(struct sc_authentic_sdo));
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2665 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 334 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 334.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 339 | 373 |
| Object | file | sdo |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
339.        struct sc_file *file = NULL;
....
373.                    sc_dump_hex(sdo->docp.acl_data, sdo-
>docp.acl_data_len));
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2666 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 334.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 373 |
| Object | file | sdo |

Code Snippet

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.         struct sc_file     *file = NULL;
```

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
373.                     sc_dump_hex(sdo->docp.acl_data, sdo-
>docp.acl_data_len));
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2667 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 334 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 334.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 339 | 373 |
| Object | file | sdo |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
339.         struct sc_file *file = NULL;
....
373.                     sc_dump_hex(sdo->docp.acl_data, sdo-
>docp.acl_data_len));
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2668 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 334.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 373 |
| Object | file | sdo |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.        struct sc_file    *file = NULL;
```

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c

Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
373.                        sc_dump_hex(sdo->docp.acl_data, sdo-
>docp.acl_data_len));
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2669 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 334 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 334.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 339 | 372 |
| Object | file | sdo |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
339.        struct sc_file *file = NULL;
....
372.        sc_log(ctx, "sdo(mech:%X,id:%X,acls:%s)", sdo->docp.mech,
sdo->docp.id,
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2670 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 334.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 372 |
| Object | file | sdo |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.         struct sc_file     *file = NULL;
```

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c

Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
372.         sc_log(ctx, "sdo(mech:%X,id:%X,acls:%s)", sdo->docp.mech,
sdo->docp.id,
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2671 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 334 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 334.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 339 | 372 |
| Object | file | sdo |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Method | authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card, |

```
....
339.        struct sc_file *file = NULL;
....
372.        sc_log(ctx, "sdo(mech:%X,id:%X,acls:%s)", sdo->docp.mech,
sdo->docp.id,
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2672 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 334.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 372 |
| Object | file | sdo |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card, |

```
....
217.        struct sc_file     *file = NULL;
```

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Method | authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card, |

```
....
372.        sc_log(ctx, "sdo(mech:%X,id:%X,acls:%s)", sdo->docp.mech,
sdo->docp.id,
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2673 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 246 |
| Object | file | file |

Code Snippet

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c

Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.         struct sc_file    *file = NULL;
....
246.         file->id = (file->id & 0xFF00) | (num & 0xFF);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2674 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 244 |
| Object | file | file |

Code Snippet

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c

Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.         struct sc_file    *file = NULL;
....
244.         sc_log(ctx, "file(type:%X), path(type:%X,path:%s)", file-
>type, file->path.type, sc_print_path(&file->path));
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20 |

| | |
|---|---|
| | 043&pathid=2675 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by file at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 244 |
| Object | file | file |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.        struct sc_file    *file = NULL;
....
244.        sc_log(ctx, "file(type:%X), path(type:%X,path:%s)", file->type, file->path.type, sc_print_path(&file->path));
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2676 |
| Status | New |

The variable declared in prkey_object at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 786 is not initialized when it is used by prkey_object at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 786.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 792 | 801 |
| Object | prkey_object | prkey_object |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_store_pubkey(struct sc_pkcs15_card *p15card, struct sc_profile *profile, struct sc_pkcs15_object *object,

```
....
792.        struct sc_pkcs15_object *prkey_object = NULL;
....
801.        prkey_info = (struct sc_pkcs15_prkey_info *)prkey_object->data;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2677 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c in line 409 is not initialized when it is used by file at OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c in line 409.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c |
| Line | 414 | 448 |
| Object | file | file |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c
Method cardos_delete_object(sc_profile_t *profile, struct sc_pkcs15_card *p15card,

```
....
414.        sc_file_t *file = NULL;
....
448.              stored_in_ef = (file->type != SC_FILE_TYPE_DF);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2678 |
| Status | New |

The variable declared in skey_obj at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 352 is not initialized when it is used by skey_obj at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 352.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Line | 405 | 416 |
| Object | skey_obj | skey_obj |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c
Method int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card,

```
....
405.                struct sc_pkcs15_object *skey_obj = NULL;
....
416.                sc_log(ctx, "found secret key '%s'", skey_obj->label);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2679 |
| Status | New |

The variable declared in skey_obj at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 352 is not initialized when it is used by skey_obj at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 352.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Line | 405 | 414 |
| Object | skey_obj | skey_obj |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Method | int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card, |

```
....
405.                struct sc_pkcs15_object *skey_obj = NULL;
....
414.                skey_info = (struct sc_pkcs15_skey_info *)skey_obj-
>data;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2680 |
| Status | New |

The variable declared in puk_info at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 575 is not initialized when it is used by puk_info at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 575.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Line | 583 | 647 |
| Object | puk_info | puk_info |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c
Method int sc_pkcs15_unblock_pin(struct sc_pkcs15_card *p15card,

```
....
583.        struct sc_pkcs15_auth_info *puk_info = NULL;
....
647.        data.pin1.pad_length = puk_info->attrs.pin.stored_length;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2681 |
| Status | New |

The variable declared in puk_info at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 575 is not initialized when it is used by puk_info at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 575.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Line | 583 | 646 |
| Object | puk_info | puk_info |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c
Method int sc_pkcs15_unblock_pin(struct sc_pkcs15_card *p15card,

```
....
583.        struct sc_pkcs15_auth_info *puk_info = NULL;
....
646.        data.pin1.max_length = puk_info->attrs.pin.max_length;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2682 |
| Status | New |

The variable declared in puk_info at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 575 is not initialized when it is used by puk_info at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 575.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Line | 583 | 645 |

| | | |
|---|---|---|
| Object | puk_info | puk_info |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c
Method int sc_pkcs15_unblock_pin(struct sc_pkcs15_card *p15card,

```
....
583.        struct sc_pkcs15_auth_info *puk_info = NULL;
....
645.        data.pin1.min_length = puk_info->attrs.pin.min_length;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2683 |
| Status | New |

The variable declared in puk_info at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 575 is not initialized when it is used by puk_info at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 575.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Line | 583 | 644 |
| Object | puk_info | puk_info |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c
Method int sc_pkcs15_unblock_pin(struct sc_pkcs15_card *p15card,

```
....
583.        struct sc_pkcs15_auth_info *puk_info = NULL;
....
644.        data.pin1.pad_char   = puk_info->attrs.pin.pad_char;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2684 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 147 is not initialized when it is used by file at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 147.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024- | OpenSC@@OpenSC-0.22.0-CVE-2024- |

| | | |
|---|---|---|
| | 1454-FP.c | 1454-FP.c |
| Line | 150 | 191 |
| Object | file | file |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c
Method authentic_pkcs15_erase_card(struct sc_profile *profile, struct sc_pkcs15_card *p15card)

```
....
150.         struct sc_file  *file = NULL;
....
191.               rv = sc_erase_binary(p15card->card, 0, file->size, 0);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2685 |
| Status | New |

The variable declared in file_p_prvkey at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 521 is not initialized when it is used by acl at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 412.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 528 | 422 |
| Object | file_p_prvkey | acl |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c
Method authentic_pkcs15_create_key(struct sc_profile *profile, struct sc_pkcs15_card *p15card,

```
....
528.         struct sc_file    *file_p_prvkey = NULL, *parent = NULL;
```

▼

File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c
Method authentic_pkcs15_fix_file_access_rule(struct sc_pkcs15_card *p15card, struct sc_file *file,

```
....
422.         acl = sc_file_get_acl_entry(file, ac_op);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2686](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2686) |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by acl at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 412.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 217 | 422 |
| Object | file | acl |

Code Snippet

File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c

Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.          struct sc_file    *file = NULL;
```

File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c

Method authentic_pkcs15_fix_file_access_rule(struct sc_pkcs15_card *p15card, struct sc_file *file,

```
....
422.          acl = sc_file_get_acl_entry(file, ac_op);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2687](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2687) |
| Status | New |

The variable declared in file_p_prvkey at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 521 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 521.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 528 | 589 |
| Object | file_p_prvkey | sdo |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_create_key(struct sc_profile *profile, struct sc_pkcs15_card *p15card, |

```
....
528.        struct sc_file    *file_p_prvkey = NULL, *parent = NULL;
....
589.        sdo->file = file_p_prvkey;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2688 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 521.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 217 | 589 |
| Object | file | sdo |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card, |

```
....
217.        struct sc_file    *file = NULL;
```

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_create_key(struct sc_profile *profile, struct sc_pkcs15_card *p15card, |

```
....
589.        sdo->file = file_p_prvkey;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2689 |
| Status | New |

The variable declared in sdo at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 521 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 521.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 526 | 592 |
| Object | sdo | sdo |

Code Snippet

File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c

Method authentic_pkcs15_create_key(struct sc_profile *profile, struct sc_pkcs15_card *p15card,

```
....
526.        struct sc_authentic_sdo *sdo = NULL;
....
592.        rv = sc_pkcs15_allocate_object_content(ctx, object,
(unsigned char *)sdo, sizeof(struct sc_authentic_sdo));
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2690 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 334 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 521.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 339 | 592 |
| Object | file | sdo |

Code Snippet

File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c

Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
339.        struct sc_file *file = NULL;
```

File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c

Method authentic_pkcs15_create_key(struct sc_profile *profile, struct sc_pkcs15_card *p15card,

```
....
592.        rv = sc_pkcs15_allocate_object_content(ctx, object,
(unsigned char *)sdo, sizeof(struct sc_authentic_sdo));
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2691 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 334 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 334.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 339 | 373 |
| Object | file | sdo |

Code Snippet

File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c

Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
339.        struct sc_file *file = NULL;
....
373.                    sc_dump_hex(sdo->docp.acl_data, sdo-
>docp.acl_data_len));
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2692 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 334.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 217 | 373 |
| Object | file | sdo |

Code Snippet

File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.            struct sc_file      *file = NULL;
```

File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c

Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
373.                        sc_dump_hex(sdo->docp.acl_data, sdo-
>docp.acl_data_len));
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2693 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 334 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 334.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 339 | 373 |
| Object | file | sdo |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c
Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
339.            struct sc_file *file = NULL;
....
373.                        sc_dump_hex(sdo->docp.acl_data, sdo-
>docp.acl_data_len));
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2694 |
| Status | New |

The variable declared in file at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by sdo at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 334.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 217 | 373 |
| Object | file | sdo |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.          struct sc_file     *file = NULL;
```

File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c
Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
373.                    sc_dump_hex(sdo->docp.acl_data, sdo-
>docp.acl_data_len));
```

## 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 | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2615 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1583 | 1624 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c

Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.       ngx_str_t                           value;
....
1624.       value.len = (size_t) sd->value_len;
```

### Use of Uninitialized Pointer\Path 2:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2616 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by data at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1583 | 1623 |
| Object | value | data |

Code Snippet

File Name openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c

Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.       ngx_str_t                           value;
....
1623.       value.data = sd->data + sd->key_len;
```

### Use of Uninitialized Pointer\Path 3:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2617 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1583 | 1626 |

| | | |
|---|---|---|
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.       ngx_str_t                     value;
....
1626.       if (*str_value_len < (size_t) value.len) {
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2618 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1583 | 1633 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.       ngx_str_t                     value;
....
1633.               *str_value_buf = malloc(value.len);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2619 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|

| | | |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1583 | 1644 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                        value;
....
1644.         *str_value_len = value.len;
```

### Use of Uninitialized Pointer\Path 6:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2620](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2620) |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by data at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1583 | 1645 |
| Object | value | data |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                        value;
....
1645.         ngx_memcpy(*str_value_buf, value.data, value.len);
```

### Use of Uninitialized Pointer\Path 7:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2621](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2621) |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1583 | 1645 |
| Object | value | len |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key, |

```
....
1583.      ngx_str_t                           value;
....
1645.          ngx_memcpy(*str_value_buf, value.data, value.len);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2622 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1583 | 1650 |
| Object | value | len |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key, |

```
....
1583.      ngx_str_t                           value;
....
1650.          if (value.len != sizeof(double)) {
```

**Use of Uninitialized Pointer\Path 9:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2623 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1583 | 1655 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                        value;
....
1655.                                &name, value.len);
```

**Use of Uninitialized Pointer\Path 10:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2624 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1583 | 1659 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                        value;
....
1659.          *str_value_len = value.len;
```

### Use of Uninitialized Pointer\Path 11:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2625 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by data at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1583 | 1660 |
| Object | value | data |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key, |

```
....
1583.      ngx_str_t                    value;
....
1660.          ngx_memcpy(num_value, value.data, sizeof(double));
```

### Use of Uninitialized Pointer\Path 12:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2626 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1583 | 1665 |
| Object | value | len |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key, |

```
....
1583.      ngx_str_t                        value;
....
1665.          if (value.len != sizeof(u_char)) {
```

### Use of Uninitialized Pointer\Path 13:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2627 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1583 | 1670 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                        value;
....
1670.                                  value.len);
```

### Use of Uninitialized Pointer\Path 14:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2628 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by data at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1583 | 1674 |
| Object | value | data |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                    value;
....
1674.          ngx_memcpy(*str_value_buf, value.data, value.len);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2629 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1583 | 1674 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                    value;
....
1674.          ngx_memcpy(*str_value_buf, value.data, value.len);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2630 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module- | openresty@@lua-nginx-module- |

| | | |
|---|---|---|
| | v0.10.18-CVE-2022-38890-FP.c | v0.10.18-CVE-2022-38890-FP.c |
| Line | 1583 | 1624 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                        value;
....
1624.      value.len = (size_t) sd->value_len;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2631 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by data at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1583 | 1623 |
| Object | value | data |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                        value;
....
1623.      value.data = sd->data + sd->key_len;
```

**Use of Uninitialized Pointer\Path 18:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2632 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1583 | 1626 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                        value;
....
1626.      if (*str_value_len < (size_t) value.len) {
```

**Use of Uninitialized Pointer\Path 19:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2633 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1583 | 1633 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                        value;
....
1633.              *str_value_buf = malloc(value.len);
```

**Use of Uninitialized Pointer\Path 20:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2634 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1583 | 1644 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                        value;
....
1644.          *str_value_len = value.len;
```

**Use of Uninitialized Pointer\Path 21:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2635 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by data at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1583 | 1645 |
| Object | value | data |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                        value;
....
1645.          ngx_memcpy(*str_value_buf, value.data, value.len);
```

### Use of Uninitialized Pointer\Path 22:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2636 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1583 | 1645 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                      value;
....
1645.         ngx_memcpy(*str_value_buf, value.data, value.len);
```

### Use of Uninitialized Pointer\Path 23:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2637 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1583 | 1650 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                          value;
....
1650.           if (value.len != sizeof(double)) {
```

### Use of Uninitialized Pointer\Path 24:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2638 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1583 | 1655 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                          value;
....
1655.                                 &name, value.len);
```

### Use of Uninitialized Pointer\Path 25:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2639 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1583 | 1659 |
| Object | value | len |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key, |

```
....
1583.      ngx_str_t                        value;
....
1659.          *str_value_len = value.len;
```

### Use of Uninitialized Pointer\Path 26:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2640 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by data at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1583 | 1660 |
| Object | value | data |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key, |

```
....
1583.      ngx_str_t                        value;
....
1660.          ngx_memcpy(num_value, value.data, sizeof(double));
```

### Use of Uninitialized Pointer\Path 27:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2641 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module- | openresty@@lua-nginx-module- |

| | | |
|---|---|---|
| | v0.10.18-CVE-2022-38890-FP.c | v0.10.18-CVE-2022-38890-FP.c |
| Line | 1583 | 1665 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                     value;
....
1665.          if (value.len != sizeof(u_char)) {
```

**Use of Uninitialized Pointer\Path 28:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2642 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by len at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1583 | 1670 |
| Object | value | len |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1583.      ngx_str_t                     value;
....
1670.                                value.len);
```

**Use of Uninitialized Pointer\Path 29:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2643 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by data at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1583 | 1674 |
| Object | value | data |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key, |

```
....
1583.      ngx_str_t                         value;
....
1674.          ngx_memcpy(*str_value_buf, value.data, value.len);
```

**Use of Uninitialized Pointer\Path 30:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2644 |
| Status | New |

The variable declared in value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573 is not initialized when it is used by value at openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c in line 1573.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1583 | 1674 |
| Object | value | value |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key, |

```
....
1583.      ngx_str_t                         value;
....
1674.          ngx_memcpy(*str_value_buf, value.data, value.len);
```

## Memory Leak

Query Path:

CPP\Cx\CPP Medium Threat\Memory Leak Version:1

Categories

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

Description

**Memory Leak\Path 1:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2602 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Line | 581 | 581 |
| Object | cache_buf | cache_buf |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c
Method static int idprime_read_binary(sc_card_t *card, unsigned int offset,

```
....
581.                    priv->cache_buf = malloc(r);
```

**Memory Leak\Path 2:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2603 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c |
| Line | 808 | 808 |
| Object | data | data |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c
Method do_cardos_extract_pubkey(sc_card_t *card, int nr, u8 tag,

```
....
808.        bn->data = malloc(count);
```

**Memory Leak\Path 3:**

| | |
|---|---|
| Severity | Medium |

| | |
|---|---|
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2604 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Line | 202 | 202 |
| Object | data | data |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Method | sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj, |

```
....
202.		obj->data = malloc(sizeof(info));
```

**Memory Leak\Path 4:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2605 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c |
| Line | 808 | 808 |
| Object | data | data |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c |
| Method | do_cardos_extract_pubkey(sc_card_t *card, int nr, u8 tag, |

```
....
808.		bn->data = malloc(count);
```

**Memory Leak\Path 5:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2606 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Line | 202 | 202 |
| Object | data | data |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Method | sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj, |

```
....
202.        obj->data = malloc(sizeof(info));
```

**Memory Leak\Path 6:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2607 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c |
| Line | 808 | 808 |
| Object | data | data |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c |
| Method | do_cardos_extract_pubkey(sc_card_t *card, int nr, u8 tag, |

```
....
808.        bn->data = malloc(count);
```

**Memory Leak\Path 7:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2608 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c |
| Line | 202 | 202 |

| | | |
|---|---|---|
| Object | data | data |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c
Method sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj,

```
....
202.        obj->data = malloc(sizeof(info));
```

**Memory Leak\Path 8:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2609 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c |
| Line | 808 | 808 |
| Object | data | data |

Code Snippet
File Name OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c
Method do_cardos_extract_pubkey(sc_card_t *card, int nr, u8 tag,

```
....
808.        bn->data = malloc(count);
```

**Memory Leak\Path 9:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2610 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c |
| Line | 202 | 202 |
| Object | data | data |

Code Snippet
File Name OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c

Method sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj,

```
....
202.         obj->data = malloc(sizeof(info));
```

**Memory Leak\Path 10:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2611 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c |
| Line | 202 | 202 |
| Object | data | data |

Code Snippet
File Name OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c
Method sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj,

```
....
202.         obj->data = malloc(sizeof(info));
```

**Memory Leak\Path 11:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2612 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c |
| Line | 202 | 202 |
| Object | data | data |

Code Snippet
File Name OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c
Method sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj,

```
....
202.          obj->data = malloc(sizeof(info));
```

**Memory Leak\Path 12:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2613 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1633 | 1633 |
| Object | str_value_buf | str_value_buf |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1633.              *str_value_buf = malloc(value.len);
```

**Memory Leak\Path 13:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2614 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1633 | 1633 |
| Object | str_value_buf | str_value_buf |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_get(ngx_shm_zone_t *zone, u_char *key,

```
....
1633.              *str_value_buf = malloc(value.len);
```

## Wrong Size t Allocation

Query Path:

CPP\Cx\CPP Integer Overflow\Wrong Size t Allocation Version:0

Description

### Wrong Size t Allocation\Path 1:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=944 |
| Status | New |

The function tlen in OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c at line 754 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c |
| Line | 774 | 774 |
| Object | tlen | tlen |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c |
| Method | static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len, |

```
....
774.        pubkey->u.rsa.modulus.data = malloc(tlen);
```

### Wrong Size t Allocation\Path 2:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=945 |
| Status | New |

The function tlen in OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c at line 754 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c |
| Line | 785 | 785 |
| Object | tlen | tlen |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c |
| Method | static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len, |

```
....
785.        pubkey->u.rsa.exponent.data = malloc(tlen);
```

### Wrong Size t Allocation\Path 3:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=946 |
| Status | New |

The function tlen in OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c at line 754 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c |
| Line | 774 | 774 |
| Object | tlen | tlen |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c |
| Method | static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len, |

```
....
774.        pubkey->u.rsa.modulus.data = malloc(tlen);
```

### Wrong Size t Allocation\Path 4:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=947 |
| Status | New |

The function tlen in OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c at line 754 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c |
| Line | 785 | 785 |
| Object | tlen | tlen |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c |

| | |
|---|---|
| Method | static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len, |

```
....
785.        pubkey->u.rsa.exponent.data = malloc(tlen);
```

**Wrong Size t Allocation\Path 5:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=948 |
| Status | New |

The function tlen in OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c at line 754 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c |
| Line | 774 | 774 |
| Object | tlen | tlen |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c |
| Method | static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len, |

```
....
774.        pubkey->u.rsa.modulus.data = malloc(tlen);
```

**Wrong Size t Allocation\Path 6:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=949 |
| Status | New |

The function tlen in OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c at line 754 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c |
| Line | 785 | 785 |
| Object | tlen | tlen |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c |
| Method | static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len, |

```
....
785.         pubkey->u.rsa.exponent.data = malloc(tlen);
```

**Wrong Size t Allocation\Path 7:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=950 |
| Status | New |

The function tlen in OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c at line 754 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c |
| Line | 774 | 774 |
| Object | tlen | tlen |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c |
| Method | static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len, |

```
....
774.         pubkey->u.rsa.modulus.data = malloc(tlen);
```

**Wrong Size t Allocation\Path 8:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=951 |
| Status | New |

The function tlen in OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c at line 754 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c |
| Line | 785 | 785 |
| Object | tlen | tlen |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c |
| Method | static int parse_ext_pubkey_file(sc_card_t *card, const u8 *data, size_t len, |

```
....
785.        pubkey->u.rsa.exponent.data = malloc(tlen);
```

**Wrong Size t Allocation\Path 9:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=952 |
| Status | New |

The function buflen in OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c at line 1047 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c |
| Line | 1090 | 1090 |
| Object | buflen | buflen |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c |
| Method | myeid_convert_ec_signature(struct sc_context *ctx, size_t s_len, unsigned char *data, size_t datalen) |

```
....
1090.       buf = calloc(1, buflen);
```

## Heap Inspection

Query Path:
CPP\Cx\CPP Medium Threat\Heap Inspection Version:1

### Categories

OWASP Top 10 2013: A6-Sensitive Data Exposure
FISMA 2014: Media Protection
NIST SP 800-53: SC-4 Information in Shared Resources (P1)
OWASP Top 10 2017: A3-Sensitive Data Exposure

*Description*

**Heap Inspection\Path 1:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2593 |
| Status | New |

Method use_certificate_chain_file at line 418 of openssl@@openssl-openssl-3.0.0-beta1-CVE-2021-3449-FP.c defines passwd_callback, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwd_callback, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | openssl@@openssl-openssl-3.0.0-beta1-CVE-2021-3449-FP.c | openssl@@openssl-openssl-3.0.0-beta1-CVE-2021-3449-FP.c |
| Line | 423 | 423 |
| Object | passwd_callback | passwd_callback |

Code Snippet

| | |
|---|---|
| File Name | openssl@@openssl-openssl-3.0.0-beta1-CVE-2021-3449-FP.c |
| Method | static int use_certificate_chain_file(SSL_CTX *ctx, SSL *ssl, const char *file) |

```
....
423.        pem_password_cb *passwd_callback;
```

**Heap Inspection\Path 2:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2594 |
| Status | New |

Method use_certificate_chain_file at line 418 of openssl@@openssl-openssl-3.0.1-CVE-2021-3449-FP.c defines passwd_callback, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwd_callback, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | openssl@@openssl-openssl-3.0.1-CVE-2021-3449-FP.c | openssl@@openssl-openssl-3.0.1-CVE-2021-3449-FP.c |
| Line | 423 | 423 |
| Object | passwd_callback | passwd_callback |

Code Snippet

| | |
|---|---|
| File Name | openssl@@openssl-openssl-3.0.1-CVE-2021-3449-FP.c |
| Method | static int use_certificate_chain_file(SSL_CTX *ctx, SSL *ssl, const char *file) |

```
....
423.        pem_password_cb *passwd_callback;
```

**Heap Inspection\Path 3:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2595 |
| Status | New |

Method use_certificate_chain_file at line 418 of openssl@@openssl-openssl-3.0.2-CVE-2021-3449-FP.c defines passwd_callback, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwd_callback, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | openssl@@openssl-openssl-3.0.2-CVE-2021-3449-FP.c | openssl@@openssl-openssl-3.0.2-CVE-2021-3449-FP.c |
| Line | 423 | 423 |
| Object | passwd_callback | passwd_callback |

Code Snippet
File Name openssl@@openssl-openssl-3.0.2-CVE-2021-3449-FP.c
Method static int use_certificate_chain_file(SSL_CTX *ctx, SSL *ssl, const char *file)

```
....
423.        pem_password_cb *passwd_callback;
```

**Heap Inspection\Path 4:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2596 |
| Status | New |

Method use_certificate_chain_file at line 418 of openssl@@openssl-openssl-3.0.4-CVE-2021-3449-FP.c defines passwd_callback, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwd_callback, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | openssl@@openssl-openssl-3.0.4-CVE-2021-3449-FP.c | openssl@@openssl-openssl-3.0.4-CVE-2021-3449-FP.c |
| Line | 423 | 423 |
| Object | passwd_callback | passwd_callback |

Code Snippet
File Name openssl@@openssl-openssl-3.0.4-CVE-2021-3449-FP.c
Method static int use_certificate_chain_file(SSL_CTX *ctx, SSL *ssl, const char *file)

```
....
423.        pem_password_cb *passwd_callback;
```

**Heap Inspection\Path 5:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2597 |
| Status | New |

Method use_certificate_chain_file at line 418 of openssl@@openssl-openssl-3.0.6-CVE-2021-3449-FP.c defines passwd_callback, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwd_callback, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | openssl@@openssl-openssl-3.0.6-CVE-2021-3449-FP.c | openssl@@openssl-openssl-3.0.6-CVE-2021-3449-FP.c |
| Line | 423 | 423 |
| Object | passwd_callback | passwd_callback |

Code Snippet

File Name openssl@@openssl-openssl-3.0.6-CVE-2021-3449-FP.c

Method static int use_certificate_chain_file(SSL_CTX *ctx, SSL *ssl, const char *file)

```
....
423.        pem_password_cb *passwd_callback;
```

**Heap Inspection\Path 6:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2598 |
| Status | New |

Method use_certificate_chain_file at line 418 of openssl@@openssl-openssl-3.0.8-CVE-2021-3449-FP.c defines passwd_callback, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwd_callback, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | openssl@@openssl-openssl-3.0.8-CVE-2021-3449-FP.c | openssl@@openssl-openssl-3.0.8-CVE-2021-3449-FP.c |
| Line | 423 | 423 |
| Object | passwd_callback | passwd_callback |

Code Snippet

File Name openssl@@openssl-openssl-3.0.8-CVE-2021-3449-FP.c

Method static int use_certificate_chain_file(SSL_CTX *ctx, SSL *ssl, const char *file)

```
....
423.        pem_password_cb *passwd_callback;
```

**Heap Inspection\Path 7:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2599 |
| Status | New |

Method use_certificate_chain_file at line 418 of openssl@@openssl-openssl-3.1.1-CVE-2021-3449-FP.c defines passwd_callback, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwd_callback, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | openssl@@openssl-openssl-3.1.1-CVE-2021-3449-FP.c | openssl@@openssl-openssl-3.1.1-CVE-2021-3449-FP.c |
| Line | 423 | 423 |
| Object | passwd_callback | passwd_callback |

Code Snippet
File Name openssl@@openssl-openssl-3.1.1-CVE-2021-3449-FP.c
Method static int use_certificate_chain_file(SSL_CTX *ctx, SSL *ssl, const char *file)

```
....
423.        pem_password_cb *passwd_callback;
```

**Heap Inspection\Path 8:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2600 |
| Status | New |

Method use_certificate_chain_file at line 437 of openssl@@openssl-openssl-3.2.0-alpha1-CVE-2021-3449-FP.c defines passwd_callback, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwd_callback, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | openssl@@openssl-openssl-3.2.0-alpha1-CVE-2021-3449-FP.c | openssl@@openssl-openssl-3.2.0-alpha1-CVE-2021-3449-FP.c |
| Line | 442 | 442 |
| Object | passwd_callback | passwd_callback |

Code Snippet
File Name openssl@@openssl-openssl-3.2.0-alpha1-CVE-2021-3449-FP.c
Method static int use_certificate_chain_file(SSL_CTX *ctx, SSL *ssl, const char *file)

```
....
442.        pem_password_cb *passwd_callback;
```

**Heap Inspection\Path 9:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2601 |
| Status | New |

Method parse_uri at line 299 of OpenSIPS@@opensips-2.4.7-CVE-2023-27597-TP.c defines pass, which is designated to contain user passwords. However, while plaintext passwords are later assigned to pass, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27597-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27597-TP.c |
| Line | 347 | 347 |
| Object | pass | pass |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-27597-TP.c |
| Method | int parse_uri(char* buf, int len, struct sip_uri* uri) |

```
....
347.        char* pass;
```

## MemoryFree on StackVariable

Query Path:

CPP\Cx\CPP Medium Threat\MemoryFree on StackVariable Version:0

Description

**MemoryFree on StackVariable\Path 1:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=926 |
| Status | New |

Calling free() (line 605) on a variable that was not dynamically allocated (line 605) in file OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 666 | 666 |
| Object | tmp | tmp |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_generate_key(struct sc_profile *profile, sc_pkcs15_card_t *p15card, |

```
....
666.        free(tmp);
```

**MemoryFree on StackVariable\Path 2:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=927 |
| Status | New |

Calling free() (line 605) on a variable that was not dynamically allocated (line 605) in file OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 666 | 666 |
| Object | tmp | tmp |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_generate_key(struct sc_profile *profile, sc_pkcs15_card_t *p15card, |

```
....
666.         free(tmp);
```

**MemoryFree on StackVariable\Path 3:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=928 |
| Status | New |

Calling free() (line 605) on a variable that was not dynamically allocated (line 605) in file OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c |
| Line | 666 | 666 |
| Object | tmp | tmp |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_generate_key(struct sc_profile *profile, sc_pkcs15_card_t *p15card, |

```
....
666.         free(tmp);
```

**MemoryFree on StackVariable\Path 4:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=929 |
| Status | New |

Calling free() (line 1858) on a variable that was not dynamically allocated (line 1858) in file OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c |
| Line | 1861 | 1861 |
| Object | priv | priv |

Code Snippet
File Name OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c
Method static int myeid_finish(sc_card_t * card)

```
....
1861.        free(priv);
```

**MemoryFree on StackVariable\Path 5:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=930 |
| Status | New |

Calling free() (line 605) on a variable that was not dynamically allocated (line 605) in file OpenSC@@OpenSC-0.23.0-rc1-CVE-2024-1454-FP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2024-1454-FP.c |
| Line | 666 | 666 |
| Object | tmp | tmp |

Code Snippet
File Name OpenSC@@OpenSC-0.23.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_generate_key(struct sc_profile *profile, sc_pkcs15_card_t *p15card,

```
....
666.        free(tmp);
```

**MemoryFree on StackVariable\Path 6:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |

| | |
|---|---|
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=931](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=931) |
| Status | New |

Calling free() (line 614) on a variable that was not dynamically allocated (line 614) in file OpenSC@@OpenSC-0.24.0-rc1-CVE-2024-1454-FP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.24.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.24.0-rc1-CVE-2024-1454-FP.c |
| Line | 675 | 675 |
| Object | tmp | tmp |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.24.0-rc1-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_generate_key(struct sc_profile *profile, sc_pkcs15_card_t *p15card, |

```
....
675.          free(tmp);
```

## Wrong Memory Allocation

Query Path:

CPP\Cx\CPP Medium Threat\Wrong Memory Allocation Version:0

### Categories

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

*Description*

**Wrong Memory Allocation\Path 1:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3293](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3293) |
| Status | New |

The function malloc in OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c at line 84 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Line | 202 | 202 |
| Object | sizeof | malloc |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |

| | |
|---|---|
| Method | sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj, |

```
....
202.         obj->data = malloc(sizeof(info));
```

**Wrong Memory Allocation\Path 2:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3294 |
| Status | New |

The function malloc in OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c at line 84 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Line | 202 | 202 |
| Object | sizeof | malloc |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Method | sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj, |

```
....
202.         obj->data = malloc(sizeof(info));
```

**Wrong Memory Allocation\Path 3:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3295 |
| Status | New |

The function malloc in OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c at line 84 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c |
| Line | 202 | 202 |
| Object | sizeof | malloc |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c |
| Method | sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj, |

```
....
202.         obj->data = malloc(sizeof(info));
```

### Wrong Memory Allocation\Path 4:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3296 |
| Status | New |

The function malloc in OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c at line 84 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c |
| Line | 202 | 202 |
| Object | sizeof | malloc |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-40660-FP.c |
| Method | sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj, |

```
....
202.         obj->data = malloc(sizeof(info));
```

### Wrong Memory Allocation\Path 5:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3297 |
| Status | New |

The function malloc in OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c at line 84 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c |

| | | |
|---|---|---|
| Line | 202 | 202 |
| Object | sizeof | malloc |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.24.0-rc1-CVE-2023-40660-FP.c |
| Method | sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj, |

```
....
202.        obj->data = malloc(sizeof(info));
```

**Wrong Memory Allocation\Path 6:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3298](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3298) |
| Status | New |

The function malloc in OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c at line 84 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c |
| Line | 202 | 202 |
| Object | sizeof | malloc |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.25.0-rc1-CVE-2023-40660-FP.c |
| Method | sc_pkcs15_decode_aodf_entry(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *obj, |

```
....
202.        obj->data = malloc(sizeof(info));
```

## Integer Overflow

Query Path:
CPP\Cx\CPP Integer Overflow\Integer Overflow Version:0

### Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.2 - Buffer overflows
FISMA 2014: System And Information Integrity
NIST SP 800-53: SI-10 Information Input Validation (P1)

*Description*

**Integer Overflow\Path 1:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=953 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 1309 of openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1500 | 1500 |
| Object | AssignExpr | AssignExpr |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ffi_shdict_store(ngx_shm_zone_t *zone, int op, u_char *key, |

```
....
1500.       n = offsetof(ngx_rbtree_node_t, color)
```

**Integer Overflow\Path 2:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=954 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 1712 of openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 1843 | 1843 |
| Object | AssignExpr | AssignExpr |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ffi_shdict_incr(ngx_shm_zone_t *zone, u_char *key, |

```
....
1843.       n = offsetof(ngx_rbtree_node_t, color)
```

**Integer Overflow\Path 3:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=955 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 1309 of openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1500 | 1500 |
| Object | AssignExpr | AssignExpr |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_store(ngx_shm_zone_t *zone, int op, u_char *key,

```
....
1500.        n = offsetof(ngx_rbtree_node_t, color)
```

**Integer Overflow\Path 4:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=956 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 1712 of openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 1843 | 1843 |
| Object | AssignExpr | AssignExpr |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c
Method ngx_http_lua_ffi_shdict_incr(ngx_shm_zone_t *zone, u_char *key,

```
....
1843.        n = offsetof(ngx_rbtree_node_t, color)
```

## Double Free

Query Path:
CPP\Cx\CPP Medium Threat\Double Free Version:1

### Categories

Description

**Double Free\Path 1:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2585 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 367 | 596 |
| Object | sdo | sdo |

Code Snippet

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c

Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
367.                  free(sdo);
```

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c

Method authentic_pkcs15_create_key(struct sc_profile *profile, struct sc_pkcs15_card *p15card,

```
....
596.            free(sdo);
```

**Double Free\Path 2:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2586 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 367 | 596 |
| Object | sdo | sdo |

Code Snippet

File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c

Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
367.                free(sdo);
```

File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c

Method authentic_pkcs15_create_key(struct sc_profile *profile, struct sc_pkcs15_card *p15card,

```
....
596.          free(sdo);
```

### Double Free\Path 3:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2587 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c |
| Line | 367 | 596 |
| Object | sdo | sdo |

Code Snippet

File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c

Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
367.                free(sdo);
```

File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c

Method authentic_pkcs15_create_key(struct sc_profile *profile, struct sc_pkcs15_card *p15card,

```
....
596.          free(sdo);
```

### Double Free\Path 4:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2588 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2024-1454-FP.c |
| Line | 367 | 596 |
| Object | sdo | sdo |

Code Snippet
File Name OpenSC@@OpenSC-0.23.0-rc1-CVE-2024-1454-FP.c
Method authentic_sdo_allocate_prvkey(struct sc_profile *profile, struct sc_card *card,

```
....
367.                    free(sdo);
```

File Name OpenSC@@OpenSC-0.23.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_create_key(struct sc_profile *profile, struct sc_pkcs15_card *p15card,

```
....
596.            free(sdo);
```

## Use of Hard coded Cryptographic Key

Query Path:
CPP\Cx\CPP Medium Threat\Use of Hard coded Cryptographic Key Version:0

### Categories

FISMA 2014: Identification And Authentication
NIST SP 800-53: SC-12 Cryptographic Key Establishment and Management (P1)
OWASP Top 10 2017: A3-Sensitive Data Exposure

*Description*

**Use of Hard coded Cryptographic Key\Path 1:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2589 |
| Status | New |

The variable key_reference at line 231 of OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c is assigned a hardcoded, literal value. This static value is used as an encryption key.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c |
| Line | 235 | 235 |
| Object | key_reference | key_reference |

Code Snippet

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-2977-TP.c
Method cardos_select_key_reference(sc_profile_t *profile, sc_pkcs15_card_t *p15card,

```
....
235.                key_info->key_reference = CARDOS_KEY_ID_MIN;
```

**Use of Hard coded Cryptographic Key\Path 2:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2590 |
| Status | New |

The variable key_reference at line 231 of OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c is assigned a hardcoded, literal value. This static value is used as an encryption key.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c |
| Line | 235 | 235 |
| Object | key_reference | key_reference |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2023-2977-TP.c
Method cardos_select_key_reference(sc_profile_t *profile, sc_pkcs15_card_t *p15card,

```
....
235.                key_info->key_reference = CARDOS_KEY_ID_MIN;
```

**Use of Hard coded Cryptographic Key\Path 3:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2591 |
| Status | New |

The variable key_reference at line 231 of OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c is assigned a hardcoded, literal value. This static value is used as an encryption key.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c |
| Line | 235 | 235 |
| Object | key_reference | key_reference |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-2977-FP.c
Method cardos_select_key_reference(sc_profile_t *profile, sc_pkcs15_card_t *p15card,

```
....
235.              key_info->key_reference = CARDOS_KEY_ID_MIN;
```

**Use of Hard coded Cryptographic Key\Path 4:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2592](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=2592) |
| Status | New |

The variable key_reference at line 231 of OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c is assigned a hardcoded, literal value. This static value is used as an encryption key.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c |
| Line | 235 | 235 |
| Object | key_reference | key_reference |

Code Snippet
File Name OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-2977-TP.c
Method cardos_select_key_reference(sc_profile_t *profile, sc_pkcs15_card_t *p15card,

```
....
235.              key_info->key_reference = CARDOS_KEY_ID_MIN;
```

## NULL Pointer Dereference

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

### Categories

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

*Description*

**NULL Pointer Dereference\Path 1:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3446](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3446) |
| Status | New |

The variable declared in null at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338 is not initialized when it is used by tag at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module- | openresty@@lua-nginx-module- |

| | | |
|---|---|---|
| | v0.10.25-CVE-2022-38890-FP.c | v0.10.25-CVE-2022-38890-FP.c |
| Line | 408 | 408 |
| Object | null | tag |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c
Method ngx_http_lua_ngx_req_set_body_data(lua_State *L)

```
....
408.                          cl->buf->tag = (ngx_buf_tag_t) NULL;
```

**NULL Pointer Dereference\Path 2:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3447 |
| Status | New |

The variable declared in null at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338 is not initialized when it is used by buf at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Line | 408 | 407 |
| Object | null | buf |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c
Method ngx_http_lua_ngx_req_set_body_data(lua_State *L)

```
....
408.                          cl->buf->tag = (ngx_buf_tag_t) NULL;
....
407.                          ngx_pfree(r->pool, cl->buf->start);
```

**NULL Pointer Dereference\Path 3:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3448 |
| Status | New |

The variable declared in null at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338 is not initialized when it is used by buf at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Line | 408 | 402 |
| Object | null | buf |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c
Method ngx_http_lua_ngx_req_set_body_data(lua_State *L)

```
....
408.                          cl->buf->tag = (ngx_buf_tag_t) NULL;
....
402.                      if (cl->buf->tag == tag && cl->buf->temporary) {
```

**NULL Pointer Dereference\Path 4:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3449 |
| Status | New |

The variable declared in null at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338 is not initialized when it is used by buf at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Line | 408 | 402 |
| Object | null | buf |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c
Method ngx_http_lua_ngx_req_set_body_data(lua_State *L)

```
....
408.                          cl->buf->tag = (ngx_buf_tag_t) NULL;
....
402.                      if (cl->buf->tag == tag && cl->buf->temporary) {
```

**NULL Pointer Dereference\Path 5:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3450 |
| Status | New |

The variable declared in null at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338 is not initialized when it is used by buf at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Line | 408 | 428 |
| Object | null | buf |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c
Method ngx_http_lua_ngx_req_set_body_data(lua_State *L)

```
....
408.                          cl->buf->tag = (ngx_buf_tag_t) NULL;
....
428.                      ngx_pfree(r->pool, cl->buf->start);
```

**NULL Pointer Dereference\Path 6:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3451 |
| Status | New |

The variable declared in null at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338 is not initialized when it is used by buf at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Line | 429 | 428 |
| Object | null | buf |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c
Method ngx_http_lua_ngx_req_set_body_data(lua_State *L)

```
....
429.                      cl->buf->tag = (ngx_buf_tag_t) NULL;
....
428.                      ngx_pfree(r->pool, cl->buf->start);
```

**NULL Pointer Dereference\Path 7:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3452 |
| Status | New |

The variable declared in null at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338 is not initialized when it is used by buf at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Line | 429 | 424 |
| Object | null | buf |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ngx_req_set_body_data(lua_State *L) |

```
....
429.                      cl->buf->tag = (ngx_buf_tag_t) NULL;
....
424.                  if (cl->buf->tag == tag && cl->buf->temporary) {
```

**NULL Pointer Dereference\Path 8:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3453 |
| Status | New |

The variable declared in null at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338 is not initialized when it is used by buf at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Line | 408 | 424 |
| Object | null | buf |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ngx_req_set_body_data(lua_State *L) |

```
....
408.                          cl->buf->tag = (ngx_buf_tag_t) NULL;
....
424.                  if (cl->buf->tag == tag && cl->buf->temporary) {
```

### NULL Pointer Dereference\Path 9:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3454 |
| Status | New |

The variable declared in null at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338 is not initialized when it is used by buf at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Line | 429 | 424 |
| Object | null | buf |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ngx_req_set_body_data(lua_State *L) |

```
....
429.                    cl->buf->tag = (ngx_buf_tag_t) NULL;
....
424.                if (cl->buf->tag == tag && cl->buf->temporary) {
```

### NULL Pointer Dereference\Path 10:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3455 |
| Status | New |

The variable declared in null at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338 is not initialized when it is used by buf at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Line | 408 | 424 |
| Object | null | buf |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ngx_req_set_body_data(lua_State *L) |

```
....
408.                    cl->buf->tag = (ngx_buf_tag_t) NULL;
....
424.                if (cl->buf->tag == tag && cl->buf->temporary) {
```

**NULL Pointer Dereference\Path 11:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3456 |
| Status | New |

The variable declared in null at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338 is not initialized when it is used by tag at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 338.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Line | 429 | 429 |
| Object | null | tag |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c
Method ngx_http_lua_ngx_req_set_body_data(lua_State *L)

```
....
429.                cl->buf->tag = (ngx_buf_tag_t) NULL;
```

**NULL Pointer Dereference\Path 12:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3457 |
| Status | New |

The variable declared in null at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 809 is not initialized when it is used by tag at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 809.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Line | 884 | 884 |
| Object | null | tag |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ngx_req_set_body_file(lua_State *L) |

```
....
884.                    cl->buf->tag = (ngx_buf_tag_t) NULL;
```

### NULL Pointer Dereference\Path 13:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3458 |
| Status | New |

The variable declared in null at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 809 is not initialized when it is used by buf at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 809.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Line | 884 | 883 |
| Object | null | buf |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_ngx_req_set_body_file(lua_State *L) |

```
....
884.                    cl->buf->tag = (ngx_buf_tag_t) NULL;
....
883.                    ngx_pfree(r->pool, cl->buf->start);
```

### NULL Pointer Dereference\Path 14:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3459 |
| Status | New |

The variable declared in null at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 809 is not initialized when it is used by buf at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 809.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Line | 884 | 879 |

| | | |
|---|---|---|
| Object | null | buf |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c
Method ngx_http_lua_ngx_req_set_body_file(lua_State *L)

```
....
884.                cl->buf->tag = (ngx_buf_tag_t) NULL;
....
879.            if (cl->buf->tag == tag && cl->buf->temporary) {
```

**NULL Pointer Dereference\Path 15:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3460 |
| Status | New |

The variable declared in null at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 809 is not initialized when it is used by buf at openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c in line 809.

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c |
| Line | 884 | 879 |
| Object | null | buf |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.25-CVE-2022-38890-FP.c
Method ngx_http_lua_ngx_req_set_body_file(lua_State *L)

```
....
884.                cl->buf->tag = (ngx_buf_tag_t) NULL;
....
879.            if (cl->buf->tag == tag && cl->buf->temporary) {
```

**NULL Pointer Dereference\Path 16:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3461 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c in line 415 is not initialized when it is used by size at OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c in line 415.

| | Source | Destination |
|---|---|---|

| | | |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c |
| Line | 419 | 435 |
| Object | null | size |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2021-42778-FP.c
Method static int idprime_get_token_name(sc_card_t* card, char** tname)

```
....
419.        sc_file_t *file = NULL;
....
435.        if (r != SC_SUCCESS || file->size == 0) {
```

**NULL Pointer Dereference\Path 17:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3462 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 338 is not initialized when it is used by len at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 352.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Line | 342 | 436 |
| Object | null | len |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c
Method _sc_pkcs15_verify_pin(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *pin_obj,

```
....
342.                    pinlen, NULL, NULL);
```

▼

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c
Method int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card,

```
....
436.                data.pin2.len = *sessionpinlen;
```

**NULL Pointer Dereference\Path 18:**

| | |
|---|---|
| Severity | Low |

| | |
|---|---|
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3463 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 338 is not initialized when it is used by pin2 at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 352.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Line | 342 | 459 |
| Object | null | pin2 |

Code Snippet

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c

Method _sc_pkcs15_verify_pin(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *pin_obj,

```
....
342.                          pinlen, NULL, NULL);
```

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c

Method int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card,

```
....
459.                          *sessionpinlen = data.pin2.len;
```

### NULL Pointer Dereference\Path 19:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3464 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 352 is not initialized when it is used by label at OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c in line 352.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c |
| Line | 405 | 416 |
| Object | null | label |

Code Snippet

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2023-40660-FP.c

Method int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card,

```
....
405.            struct sc_pkcs15_object *skey_obj = NULL;
....
416.            sc_log(ctx, "found secret key '%s'", skey_obj->label);
```

### NULL Pointer Dereference\Path 20:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3465 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by path at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 253 |
| Object | null | path |

Code Snippet

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c

Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.        struct sc_file     *file = NULL;
....
253.            file->path.value[file->path.len - 1] = file->id &
0xFF;
```

### NULL Pointer Dereference\Path 21:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3466 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by path at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 252 |
| Object | null | path |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card, |

```
....
217.        struct sc_file     *file = NULL;
....
252.              file->path.value[file->path.len - 2] = (file->id >> 8)
& 0xFF;
```

**NULL Pointer Dereference\Path 22:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3467 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by path at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 248 |
| Object | null | path |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card, |

```
....
217.        struct sc_file     *file = NULL;
....
248.              if (file->path.len == 0)    {
```

**NULL Pointer Dereference\Path 23:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3468 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by path at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |

| | | |
|---|---|---|
| Line | 217 | 244 |
| Object | null | path |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.        struct sc_file    *file = NULL;
....
244.        sc_log(ctx, "file(type:%X), path(type:%X,path:%s)", file-
>type, file->path.type, sc_print_path(&file->path));
```

**NULL Pointer Dereference\Path 24:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3469 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by path at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 244 |
| Object | null | path |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.        struct sc_file    *file = NULL;
....
244.        sc_log(ctx, "file(type:%X), path(type:%X,path:%s)", file-
>type, file->path.type, sc_print_path(&file->path));
```

**NULL Pointer Dereference\Path 25:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3470 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by type at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 244 |
| Object | null | type |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.        struct sc_file     *file = NULL;
....
244.        sc_log(ctx, "file(type:%X), path(type:%X,path:%s)", file-
>type, file->path.type, sc_print_path(&file->path));
```

### NULL Pointer Dereference\Path 26:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3471 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 412 is not initialized when it is used by auth_id at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 381.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 429 | 389 |
| Object | null | auth_id |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_fix_file_access_rule(struct sc_pkcs15_card *p15card, struct sc_file *file,

```
....
429.                rv = authentic_pkcs15_add_access_rule(object,
rule_mode, NULL);
```

▼

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_add_access_rule(struct sc_pkcs15_object *object, unsigned access_mode, struct sc_pkcs15_id *auth_id)

```
....
389.                              object->access_rules[ii].auth_id =
*auth_id;
```

### NULL Pointer Dereference\Path 27:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3472 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 786 is not initialized when it is used by auth_id at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 381.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 816 | 389 |
| Object | null | auth_id |

Code Snippet

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c

Method authentic_store_pubkey(struct sc_pkcs15_card *p15card, struct sc_profile *profile, struct sc_pkcs15_object *object,

```
....
816.         authentic_pkcs15_add_access_rule(object,
SC_PKCS15_ACCESS_RULE_MODE_READ, NULL);
```

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c

Method authentic_pkcs15_add_access_rule(struct sc_pkcs15_object *object, unsigned access_mode, struct sc_pkcs15_id *auth_id)

```
....
389.                              object->access_rules[ii].auth_id =
*auth_id;
```

### NULL Pointer Dereference\Path 28:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3473 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 786 is not initialized when it is used by auth_id at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 381.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 816 | 394 |
| Object | null | auth_id |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_store_pubkey(struct sc_pkcs15_card *p15card, struct sc_profile *profile, struct sc_pkcs15_object *object,

```
....
816.         authentic_pkcs15_add_access_rule(object, 
SC_PKCS15_ACCESS_RULE_MODE_READ, NULL);
```

File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_add_access_rule(struct sc_pkcs15_object *object, unsigned access_mode, struct sc_pkcs15_id *auth_id)

```
....
394.                  else if (!auth_id && !object-
>access_rules[ii].auth_id.len)    {
```

**NULL Pointer Dereference\Path 29:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3474 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 412 is not initialized when it is used by auth_id at OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c in line 381.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Line | 429 | 394 |
| Object | null | auth_id |

Code Snippet
File Name OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_fix_file_access_rule(struct sc_pkcs15_card *p15card, struct sc_file *file,

```
....
429.                  rv = authentic_pkcs15_add_access_rule(object,
rule_mode, NULL);
```

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.21.0-rc1-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_add_access_rule(struct sc_pkcs15_object *object, unsigned access_mode, struct sc_pkcs15_id *auth_id) |

```
....
394.                  else if (!auth_id && !object-
>access_rules[ii].auth_id.len)    {
```

### NULL Pointer Dereference\Path 30:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3475](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3475) |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 338 is not initialized when it is used by len at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 352.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Line | 342 | 436 |
| Object | null | len |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Method | _sc_pkcs15_verify_pin(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *pin_obj, |

```
....
342.                        pinlen, NULL, NULL);
```

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Method | int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card, |

```
....
436.                  data.pin2.len = *sessionpinlen;
```

### NULL Pointer Dereference\Path 31:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3476 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 338 is not initialized when it is used by pin2 at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 352.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Line | 342 | 459 |
| Object | null | pin2 |

Code Snippet

File Name OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c

Method _sc_pkcs15_verify_pin(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *pin_obj,

```
....
342.                    pinlen, NULL, NULL);
```

File Name OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c

Method int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card,

```
....
459.                    *sessionpinlen = data.pin2.len;
```

### NULL Pointer Dereference\Path 32:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3477 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 352 is not initialized when it is used by label at OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c in line 352.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c |
| Line | 405 | 416 |
| Object | null | label |

Code Snippet

File Name OpenSC@@OpenSC-0.22.0-CVE-2023-40660-TP.c

Method int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card,

```
....
405.                struct sc_pkcs15_object *skey_obj = NULL;
....
416.                sc_log(ctx, "found secret key '%s'", skey_obj->label);
```

### NULL Pointer Dereference\Path 33:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3478 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by path at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 217 | 253 |
| Object | null | path |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.          struct sc_file    *file = NULL;
....
253.                file->path.value[file->path.len - 1] = file->id &
0xFF;
```

### NULL Pointer Dereference\Path 34:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3479 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by path at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 217 | 252 |
| Object | null | path |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card, |

```
....
217.        struct sc_file     *file = NULL;
....
252.              file->path.value[file->path.len - 2] = (file->id >> 8)
& 0xFF;
```

### NULL Pointer Dereference\Path 35:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3480 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by path at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 217 | 248 |
| Object | null | path |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card, |

```
....
217.        struct sc_file     *file = NULL;
....
248.              if (file->path.len == 0)    {
```

### NULL Pointer Dereference\Path 36:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3481 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by path at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |

| | | |
|---|---|---|
| Line | 217 | 244 |
| Object | null | path |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.         struct sc_file    *file = NULL;
....
244.         sc_log(ctx, "file(type:%X), path(type:%X,path:%s)", file-
>type, file->path.type, sc_print_path(&file->path));
```

### NULL Pointer Dereference\Path 37:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3482 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by path at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 217 | 244 |
| Object | null | path |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.         struct sc_file    *file = NULL;
....
244.         sc_log(ctx, "file(type:%X), path(type:%X,path:%s)", file-
>type, file->path.type, sc_print_path(&file->path));
```

### NULL Pointer Dereference\Path 38:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3483 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by type at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 217 | 244 |
| Object | null | type |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.        struct sc_file     *file = NULL;
....
244.        sc_log(ctx, "file(type:%X), path(type:%X,path:%s)", file-
>type, file->path.type, sc_print_path(&file->path));
```

**NULL Pointer Dereference\Path 39:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3484 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 412 is not initialized when it is used by auth_id at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 381.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 429 | 389 |
| Object | null | auth_id |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c
Method authentic_pkcs15_fix_file_access_rule(struct sc_pkcs15_card *p15card, struct sc_file *file,

```
....
429.                rv = authentic_pkcs15_add_access_rule(object,
rule_mode, NULL);
```

▼

File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c
Method authentic_pkcs15_add_access_rule(struct sc_pkcs15_object *object, unsigned access_mode, struct sc_pkcs15_id *auth_id)

```
....
389.                          object->access_rules[ii].auth_id =
*auth_id;
```

### NULL Pointer Dereference\Path 40:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3485 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 786 is not initialized when it is used by auth_id at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 381.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 816 | 389 |
| Object | null | auth_id |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c
Method authentic_store_pubkey(struct sc_pkcs15_card *p15card, struct sc_profile *profile, struct sc_pkcs15_object *object,

```
....
816.        authentic_pkcs15_add_access_rule(object,
SC_PKCS15_ACCESS_RULE_MODE_READ, NULL);
```

File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c
Method authentic_pkcs15_add_access_rule(struct sc_pkcs15_object *object, unsigned access_mode, struct sc_pkcs15_id *auth_id)

```
....
389.                          object->access_rules[ii].auth_id =
*auth_id;
```

### NULL Pointer Dereference\Path 41:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3486 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 412 is not initialized when it is used by auth_id at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 381.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 429 | 394 |
| Object | null | auth_id |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_fix_file_access_rule(struct sc_pkcs15_card *p15card, struct sc_file *file, |

```
....
429.                 rv = authentic_pkcs15_add_access_rule(object, 
rule_mode, NULL);
```

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Method | authentic_pkcs15_add_access_rule(struct sc_pkcs15_object *object, unsigned access_mode, struct sc_pkcs15_id *auth_id) |

```
....
394.                 else if (!auth_id && !object-
>access_rules[ii].auth_id.len)    {
```

**NULL Pointer Dereference\Path 42:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3487 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 786 is not initialized when it is used by auth_id at OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c in line 381.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Line | 816 | 394 |
| Object | null | auth_id |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c |
| Method | authentic_store_pubkey(struct sc_pkcs15_card *p15card, struct sc_profile *profile, struct sc_pkcs15_object *object, |

```
....
816.        authentic_pkcs15_add_access_rule(object,
SC_PKCS15_ACCESS_RULE_MODE_READ, NULL);
```

File Name OpenSC@@OpenSC-0.22.0-CVE-2024-1454-FP.c

Method authentic_pkcs15_add_access_rule(struct sc_pkcs15_object *object, unsigned access_mode, struct sc_pkcs15_id *auth_id)

```
....
394.              else if (!auth_id && !object-
>access_rules[ii].auth_id.len)    {
```

### NULL Pointer Dereference\Path 43:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3488 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c in line 338 is not initialized when it is used by len at OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c in line 352.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c |
| Line | 342 | 436 |
| Object | null | len |

Code Snippet

File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c

Method _sc_pkcs15_verify_pin(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *pin_obj,

```
....
342.                    pinlen, NULL, NULL);
```

File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c

Method int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card,

```
....
436.              data.pin2.len = *sessionpinlen;
```

### NULL Pointer Dereference\Path 44:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3489 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c in line 338 is not initialized when it is used by pin2 at OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c in line 352.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c |
| Line | 342 | 459 |
| Object | null | pin2 |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c |
| Method | _sc_pkcs15_verify_pin(struct sc_pkcs15_card *p15card, struct sc_pkcs15_object *pin_obj, |

```
....
342.                        pinlen, NULL, NULL);
```

▼

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c |
| Method | int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card, |

```
....
459.                        *sessionpinlen = data.pin2.len;
```

**NULL Pointer Dereference\Path 45:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3490 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c in line 352 is not initialized when it is used by label at OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c in line 352.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c |
| Line | 405 | 416 |
| Object | null | label |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.22.0-rc1-CVE-2023-40660-TP.c |
| Method | int sc_pkcs15_verify_pin_with_session_pin(struct sc_pkcs15_card *p15card, |

```
....
405.                struct sc_pkcs15_object *skey_obj = NULL;
....
416.                sc_log(ctx, "found secret key '%s'", skey_obj->label);
```

### NULL Pointer Dereference\Path 46:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3491 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by path at OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 253 |
| Object | null | path |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.          struct sc_file     *file = NULL;
....
253.                file->path.value[file->path.len - 1] = file->id &
0xFF;
```

### NULL Pointer Dereference\Path 47:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3492 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by path at OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 252 |
| Object | null | path |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.        struct sc_file     *file = NULL;
....
252.              file->path.value[file->path.len - 2] = (file->id >> 8)
& 0xFF;
```

**NULL Pointer Dereference\Path 48:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3493 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by path at OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 248 |
| Object | null | path |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.        struct sc_file     *file = NULL;
....
248.              if (file->path.len == 0)    {
```

**NULL Pointer Dereference\Path 49:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3494 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by path at OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c |

| | | |
|---|---|---|
| Line | 217 | 244 |
| Object | null | path |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.        struct sc_file    *file = NULL;
....
244.        sc_log(ctx, "file(type:%X), path(type:%X,path:%s)", file->type, file->path.type, sc_print_path(&file->path));
```

**NULL Pointer Dereference\Path 50:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3495 |
| Status | New |

The variable declared in null at OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c in line 213 is not initialized when it is used by path at OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c in line 213.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c | OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c |
| Line | 217 | 244 |
| Object | null | path |

Code Snippet
File Name OpenSC@@OpenSC-0.22.0-rc1-CVE-2024-1454-FP.c
Method authentic_pkcs15_new_file(struct sc_profile *profile, struct sc_card *card,

```
....
217.        struct sc_file    *file = NULL;
....
244.        sc_log(ctx, "file(type:%X), path(type:%X,path:%s)", file->type, file->path.type, sc_print_path(&file->path));
```

## Unchecked Array Index

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

Categories

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

*Description*

**Unchecked Array Index\Path 1:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4049 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c |
| Line | 1346 | 1346 |
| Object | crgram_half | crgram_half |

Code Snippet

File Name OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c

Method static int myeid_transmit_decipher_pi_split(struct sc_card *card, struct sc_apdu *apdu, u8 *sbuf)

```
....
1346.        sbuf[crgram_half] = 0x82;      /* Padding Indicator, 0x82 =
Second half */
```

**Unchecked Array Index\Path 2:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4050 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27596-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27596-TP.c |
| Line | 122 | 122 |
| Object | count | count |

Code Snippet

File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27596-TP.c

Method static int create_codec_lumps(struct sip_msg * msg)

```
....
122.                    lumps[count] = 1;
```

**Unchecked Array Index\Path 3:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4051 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27596-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27596-TP.c |
| Line | 470 | 470 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27596-TP.c
Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
470.                    payload->rtp_enc.s[payload->rtp_enc.len] =
0;
```

**Unchecked Array Index\Path 4:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4052 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27596-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27596-TP.c |
| Line | 472 | 472 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27596-TP.c
Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
472.                    payload->rtp_enc.s[payload->rtp_enc.len] =
temp;
```

**Unchecked Array Index\Path 5:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4053 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27600-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27600-TP.c |

|  |  |  |
|---|---|---|
| Line | 122 | 122 |
| Object | count | count |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27600-TP.c
Method static int create_codec_lumps(struct sip_msg * msg)

```
....
122.                     lumps[count] = 1;
```

**Unchecked Array Index\Path 6:**

|  |  |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4054 |
| Status | New |

|  | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27600-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27600-TP.c |
| Line | 470 | 470 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27600-TP.c
Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
470.                         payload->rtp_enc.s[payload->rtp_enc.len] = 
0;
```

**Unchecked Array Index\Path 7:**

|  |  |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4055 |
| Status | New |

|  | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27600-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27600-TP.c |
| Line | 472 | 472 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27600-TP.c

Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
472.                                payload->rtp_enc.s[payload->rtp_enc.len] =
temp;
```

**Unchecked Array Index\Path 8:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4056 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27601-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27601-TP.c |
| Line | 122 | 122 |
| Object | count | count |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27601-TP.c
Method static int create_codec_lumps(struct sip_msg * msg)

```
....
122.                          lumps[count] = 1;
```

**Unchecked Array Index\Path 9:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4057 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27601-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27601-TP.c |
| Line | 470 | 470 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27601-TP.c
Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
470.                                payload->rtp_enc.s[payload->rtp_enc.len] =
0;
```

### Unchecked Array Index\Path 10:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4058 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27601-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27601-TP.c |
| Line | 472 | 472 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27601-TP.c
Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
472.                            payload->rtp_enc.s[payload->rtp_enc.len] =
temp;
```

### Unchecked Array Index\Path 11:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4059 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c |
| Line | 262 | 262 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c
Method char* received_builder(struct sip_msg *msg, unsigned int *received_len)

```
....
262.        buf[len]=0; /*null terminate it */
```

### Unchecked Array Index\Path 12:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20 |

| | |
|---|---|
| | 043&pathid=4060 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c |
| Line | 286 | 286 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c
Method char* rport_builder(struct sip_msg *msg, unsigned int *rport_len)

```
....
286.        buf[len]=0; /*null terminate it*/
```

**Unchecked Array Index\Path 13:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4061 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c |
| Line | 316 | 316 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c
Method char* id_builder(struct sip_msg* msg, unsigned int *id_len)

```
....
316.        buf[len]=0; /* null terminate it */
```

**Unchecked Array Index\Path 14:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4062 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c |

| | | |
|---|---|---|
| Line | 348 | 348 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c
Method char* clen_builder(struct sip_msg* msg, int *clen_len, int diff)

```
....
348.         buf[len]=0; /* null terminate it */
```

### Unchecked Array Index\Path 15:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4063 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c |
| Line | 2820 | 2820 |
| Object | via_len | via_len |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c
Method char* via_builder( unsigned int *len,

```
....
2820.        line_buf[via_len]=':'; via_len++;
```

### Unchecked Array Index\Path 16:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4064 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c |
| Line | 2839 | 2839 |
| Object | via_len | via_len |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c

Method char* via_builder( unsigned int *len,

```
....
2839.       line_buf[via_len]=0; /* null terminate the string*/
```

### Unchecked Array Index\Path 17:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4065 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c |
| Line | 2898 | 2898 |
| Object | pos | pos |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28095-TP.c
Method char *construct_uri(str *protocol,str *username,str *domain,str *port,

```
....
2898.       uri_buff[pos] = 0;
```

### Unchecked Array Index\Path 18:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4066 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 728 | 728 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
728.          ptr2[len] = '\"';
```

### Unchecked Array Index\Path 19:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4067 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c |
| Line | 256 | 256 |
| Object | len | len |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c |
| Method | int add_dest2list(int id, str uri, struct socket_info *sock, str *comsock, int state, |

```
....
256.            dp->dst_uri.s[dp->dst_uri.len]='\0';
```

### Unchecked Array Index\Path 20:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4068 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c |
| Line | 580 | 580 |
| Object | end_idx | end_idx |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c |
| Method | int ds_pvar_algo(struct sip_msg *msg, ds_set_p set, ds_dest_p **sorted_set, |

```
....
580.            sset[end_idx] = &set->dlist[end_idx];
```

### Unchecked Array Index\Path 21:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4069 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27596-TP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27596-TP.c |
| Line | 105 | 105 |
| Object | count | count |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27596-TP.c |
| Method | static int create_codec_lumps(struct sip_msg * msg) |

```
....
105.                    lumps[count] = 1;
```

**Unchecked Array Index\Path 22:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4070 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27596-TP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27596-TP.c |
| Line | 453 | 453 |
| Object | len | len |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27596-TP.c |
| Method | static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell, |

```
....
453.                    payload->rtp_enc.s[payload->rtp_enc.len] =
0;
```

**Unchecked Array Index\Path 23:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4071 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27596-TP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27596-TP.c |
| Line | 455 | 455 |

| | | |
|---|---|---|
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27596-TP.c
Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
455.                    payload->rtp_enc.s[payload->rtp_enc.len] =
temp;
```

**Unchecked Array Index\Path 24:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4072 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27600-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27600-FP.c |
| Line | 105 | 105 |
| Object | count | count |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27600-FP.c
Method static int create_codec_lumps(struct sip_msg * msg)

```
....
105.                    lumps[count] = 1;
```

**Unchecked Array Index\Path 25:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4073 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27600-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27600-FP.c |
| Line | 453 | 453 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27600-FP.c
Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
453.                          payload->rtp_enc.s[payload->rtp_enc.len] =
0;
```

**Unchecked Array Index\Path 26:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4074 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27600-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27600-FP.c |
| Line | 455 | 455 |
| Object | len | len |

Code Snippet

File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27600-FP.c

Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
455.                          payload->rtp_enc.s[payload->rtp_enc.len] =
temp;
```

**Unchecked Array Index\Path 27:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4075 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27601-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27601-FP.c |
| Line | 105 | 105 |
| Object | count | count |

Code Snippet

File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27601-FP.c

Method static int create_codec_lumps(struct sip_msg * msg)

```
....
105.                     lumps[count] = 1;
```

### Unchecked Array Index\Path 28:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4076 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27601-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27601-FP.c |
| Line | 453 | 453 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27601-FP.c
Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
453.                              payload->rtp_enc.s[payload->rtp_enc.len] =
0;
```

### Unchecked Array Index\Path 29:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4077 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27601-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27601-FP.c |
| Line | 455 | 455 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27601-FP.c
Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
455.                              payload->rtp_enc.s[payload->rtp_enc.len] =
temp;
```

### Unchecked Array Index\Path 30:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20 |

| | |
|---|---|
| | 043&pathid=4078 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c |
| Line | 262 | 262 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c
Method char* received_builder(struct sip_msg *msg, unsigned int *received_len)

```
....
262.        buf[len]=0; /*null terminate it */
```

**Unchecked Array Index\Path 31:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4079 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c |
| Line | 286 | 286 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c
Method char* rport_builder(struct sip_msg *msg, unsigned int *rport_len)

```
....
286.        buf[len]=0; /*null terminate it*/
```

**Unchecked Array Index\Path 32:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4080 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c |

| | | |
|---|---|---|
| Line | 316 | 316 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c
Method char* id_builder(struct sip_msg* msg, unsigned int *id_len)

```
....
316.        buf[len]=0; /* null terminate it */
```

**Unchecked Array Index\Path 33:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4081 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c |
| Line | 348 | 348 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c
Method char* clen_builder(struct sip_msg* msg, int *clen_len, int diff)

```
....
348.        buf[len]=0; /* null terminate it */
```

**Unchecked Array Index\Path 34:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4082 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c |
| Line | 2833 | 2833 |
| Object | via_len | via_len |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c

Method char* via_builder( unsigned int *len,

```
....
2833.        line_buf[via_len]=':'; via_len++;
```

### Unchecked Array Index\Path 35:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4083 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c |
| Line | 2852 | 2852 |
| Object | via_len | via_len |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c
Method char* via_builder( unsigned int *len,

```
....
2852.        line_buf[via_len]=0; /* null terminate the string*/
```

### Unchecked Array Index\Path 36:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4084 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c |
| Line | 2911 | 2911 |
| Object | pos | pos |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28095-TP.c
Method char *construct_uri(str *protocol,str *username,str *domain,str *port,

```
....
2911.        uri_buff[pos] = 0;
```

### Unchecked Array Index\Path 37:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4085 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 751 | 751 |
| Object | len | len |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Method | static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p) |

```
....
751.          ptr2[len] = '\"';
```

**Unchecked Array Index\Path 38:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4086 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c |
| Line | 240 | 240 |
| Object | len | len |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c |
| Method | int add_dest2list(int id, str uri, struct socket_info *sock, str *comsock, int state, |

```
....
240.             dp->dst_uri.s[dp->dst_uri.len]='\0';
```

**Unchecked Array Index\Path 39:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4087 |
| Status | New |

|  | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c |
| Line | 564 | 564 |
| Object | end_idx | end_idx |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c
Method int ds_pvar_algo(struct sip_msg *msg, ds_set_p set, ds_dest_p **sorted_set,

```
....
564.            sset[end_idx] = &set->dlist[end_idx];
```

**Unchecked Array Index\Path 40:**

|  |  |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4088 |
| Status | New |

|  | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c |
| Line | 722 | 722 |
| Object | end_idx | end_idx |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c
Method int ds_route_algo(struct sip_msg *msg, ds_set_p set,

```
....
722.            sset[end_idx] = &set->dlist[end_idx];
```

**Unchecked Array Index\Path 41:**

|  |  |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4089 |
| Status | New |

|  | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27596-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27596-TP.c |
| Line | 105 | 105 |

| | | |
|---|---|---|
| Object | count | count |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27596-TP.c
Method static int create_codec_lumps(struct sip_msg * msg)

```
....
105.                    lumps[count] = 1;
```

**Unchecked Array Index\Path 42:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4090 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27596-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27596-TP.c |
| Line | 453 | 453 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27596-TP.c
Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
453.                            payload->rtp_enc.s[payload->rtp_enc.len] =
0;
```

**Unchecked Array Index\Path 43:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4091 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27596-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27596-TP.c |
| Line | 455 | 455 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27596-TP.c
Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
455.                          payload->rtp_enc.s[payload->rtp_enc.len] =
temp;
```

### Unchecked Array Index\Path 44:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4092 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27600-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27600-TP.c |
| Line | 105 | 105 |
| Object | count | count |

Code Snippet

File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27600-TP.c

Method static int create_codec_lumps(struct sip_msg * msg)

```
....
105.                    lumps[count] = l;
```

### Unchecked Array Index\Path 45:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4093 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27600-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27600-TP.c |
| Line | 453 | 453 |
| Object | len | len |

Code Snippet

File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27600-TP.c

Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
453.                          payload->rtp_enc.s[payload->rtp_enc.len] =
0;
```

### Unchecked Array Index\Path 46:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4094 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27600-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27600-TP.c |
| Line | 455 | 455 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27600-TP.c
Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
455.                              payload->rtp_enc.s[payload->rtp_enc.len] =
temp;
```

### Unchecked Array Index\Path 47:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4095 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27601-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27601-TP.c |
| Line | 105 | 105 |
| Object | count | count |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27601-TP.c
Method static int create_codec_lumps(struct sip_msg * msg)

```
....
105.                        lumps[count] = 1;
```

### Unchecked Array Index\Path 48:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4096 |

| | |
|---|---|
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27601-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27601-TP.c |
| Line | 453 | 453 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27601-TP.c
Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
453.                             payload->rtp_enc.s[payload->rtp_enc.len] =
0;
```

**Unchecked Array Index\Path 49:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4097 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27601-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27601-TP.c |
| Line | 455 | 455 |
| Object | len | len |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27601-TP.c
Method static int stream_process(struct sip_msg * msg, struct sdp_stream_cell *cell,

```
....
455.                             payload->rtp_enc.s[payload->rtp_enc.len] =
temp;
```

**Unchecked Array Index\Path 50:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=4098 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023- | OpenSIPS@@opensips-3.1.1-CVE-2023- |

| | | |
|---|---|---|
| | 28095-TP.c | 28095-TP.c |
| Line | 262 | 262 |
| Object | len | len |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.1-CVE-2023-28095-TP.c |
| Method | char* received_builder(struct sip_msg *msg, unsigned int *received_len) |

```
....
262.        buf[len]=0; /*null terminate it */
```

## Use of Sizeof On a Pointer Type

Query Path:
CPP\Cx\CPP Low Visibility\Use of Sizeof On a Pointer Type Version:1

Description

**Use of Sizeof On a Pointer Type\Path 1:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3374 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c |
| Line | 200 | 200 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c |
| Method | ngx_http_lua_ngx_location_capture_multi(lua_State *L) |

```
....
200.      sr_headers_len  = nsubreqs * sizeof(ngx_http_headers_out_t *);
```

**Use of Sizeof On a Pointer Type\Path 2:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3375 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c |
| Line | 543 | 543 |

| | | |
|---|---|---|
| Object | sizeof | sizeof |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c
Method ngx_http_lua_ngx_location_capture_multi(lua_State *L)

```
....
543.            ofs1 = ngx_align(sizeof(ngx_http_post_subrequest_t),
sizeof(void *));
```

### Use of Sizeof On a Pointer Type\Path 3:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3376 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c |
| Line | 544 | 544 |
| Object | sizeof | sizeof |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c
Method ngx_http_lua_ngx_location_capture_multi(lua_State *L)

```
....
544.            ofs2 = ngx_align(sizeof(ngx_http_lua_ctx_t), sizeof(void
*));
```

### Use of Sizeof On a Pointer Type\Path 4:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3377 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c |
| Line | 559 | 559 |
| Object | sizeof | sizeof |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c

| | |
|---|---|
| Method | ngx_http_lua_ngx_location_capture_multi(lua_State *L) |

```
....
559.
sizeof(void *)));
```

**Use of Sizeof On a Pointer Type\Path 5:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3378 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c |
| Line | 566 | 566 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c |
| Method | ngx_http_lua_ngx_location_capture_multi(lua_State *L) |

```
....
566.
sizeof(void *)));
```

**Use of Sizeof On a Pointer Type\Path 6:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3379 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c |
| Line | 1501 | 1501 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.16-CVE-2020-11724-TP.c |
| Method | ngx_http_lua_subrequest(ngx_http_request_t *r, |

```
....
1501.       sr->ctx = ngx_pcalloc(r->pool, sizeof(void *) *
ngx_http_max_module);
```

### Use of Sizeof On a Pointer Type\Path 7:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3380 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Line | 356 | 356 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.16-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_inject_shdict_api(ngx_http_lua_main_conf_t *lmcf, lua_State *L) |

```
....
356.                zone_udata = lua_newuserdata(L, sizeof(ngx_shm_zone_t
*));
```

### Use of Sizeof On a Pointer Type\Path 8:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3381 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Line | 356 | 356 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | openresty@@lua-nginx-module-v0.10.18-CVE-2022-38890-FP.c |
| Method | ngx_http_lua_inject_shdict_api(ngx_http_lua_main_conf_t *lmcf, lua_State *L) |

```
....
356.                zone_udata = lua_newuserdata(L, sizeof(ngx_shm_zone_t
*));
```

### Use of Sizeof On a Pointer Type\Path 9:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3382 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c |
| Line | 198 | 198 |
| Object | sizeof | sizeof |

| | |
|---|---|
| Code Snippet | |
| File Name | openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c |
| Method | ngx_http_lua_ngx_location_capture_multi(lua_State *L) |

```
....
198.      sr_headers_len  = nsubreqs * sizeof(ngx_http_headers_out_t *);
```

### Use of Sizeof On a Pointer Type\Path 10:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3383 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c |
| Line | 541 | 541 |
| Object | sizeof | sizeof |

| | |
|---|---|
| Code Snippet | |
| File Name | openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c |
| Method | ngx_http_lua_ngx_location_capture_multi(lua_State *L) |

```
....
541.          ofs1 = ngx_align(sizeof(ngx_http_post_subrequest_t),
sizeof(void *));
```

### Use of Sizeof On a Pointer Type\Path 11:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20 |

| | |
|---|---|
| | 043&pathid=3384 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c |
| Line | 542 | 542 |
| Object | sizeof | sizeof |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c
Method ngx_http_lua_ngx_location_capture_multi(lua_State *L)

```
....
542.            ofs2 = ngx_align(sizeof(ngx_http_lua_ctx_t), sizeof(void
*));
```

**Use of Sizeof On a Pointer Type\Path 12:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3385 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c |
| Line | 557 | 557 |
| Object | sizeof | sizeof |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c
Method ngx_http_lua_ngx_location_capture_multi(lua_State *L)

```
....
557.
sizeof(void *)));
```

**Use of Sizeof On a Pointer Type\Path 13:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3386 |
| Status | New |

| | Source | Destination |
|---|---|---|

|  |  |  |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c |
| Line | 564 | 564 |
| Object | sizeof | sizeof |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c
Method ngx_http_lua_ngx_location_capture_multi(lua_State *L)

```
....
564.
sizeof(void *)));
```

### Use of Sizeof On a Pointer Type\Path 14:

|  |  |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3387 |
| Status | New |

|  | Source | Destination |
|---|---|---|
| File | openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c | openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c |
| Line | 1391 | 1391 |
| Object | sizeof | sizeof |

Code Snippet
File Name openresty@@lua-nginx-module-v0.10.25-CVE-2020-11724-TP.c
Method ngx_http_lua_subrequest(ngx_http_request_t *r,

```
....
1391.       sr->ctx = ngx_pcalloc(r->pool, sizeof(void *) *
ngx_http_max_module);
```

### Use of Sizeof On a Pointer Type\Path 15:

|  |  |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3388 |
| Status | New |

|  | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27596-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27596-TP.c |
| Line | 101 | 101 |

| | | |
|---|---|---|
| Object | sizeof | sizeof |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27596-TP.c
Method static int create_codec_lumps(struct sip_msg * msg)

```
....
101.        memset(lumps, 0, MAX_STREAMS * sizeof(struct lump*));
```

**Use of Sizeof On a Pointer Type\Path 16:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3389](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3389) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27600-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27600-TP.c |
| Line | 101 | 101 |
| Object | sizeof | sizeof |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27600-TP.c
Method static int create_codec_lumps(struct sip_msg * msg)

```
....
101.        memset(lumps, 0, MAX_STREAMS * sizeof(struct lump*));
```

**Use of Sizeof On a Pointer Type\Path 17:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3390](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3390) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-27601-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-27601-TP.c |
| Line | 101 | 101 |
| Object | sizeof | sizeof |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-27601-TP.c
Method static int create_codec_lumps(struct sip_msg * msg)

```
....
101.        memset(lumps, 0, MAX_STREAMS * sizeof(struct lump*));
```

**Use of Sizeof On a Pointer Type\Path 18:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3391](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3391) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 1262 | 1262 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_array(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p) |

```
....
1262.          entries = (unsigned char**)cJSON_malloc(numentries * 
sizeof(unsigned char*));
```

**Use of Sizeof On a Pointer Type\Path 19:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3392](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3392) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 1267 | 1267 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_array(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p) |

```
....
1267.          memset(entries, '\0', numentries * sizeof(unsigned
char*));
```

### Use of Sizeof On a Pointer Type\Path 20:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3393 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 1602 | 1602 |
| Object | sizeof | sizeof |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c
Method static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1602.          entries = (unsigned char**)cJSON_malloc(numentries *
sizeof(unsigned char*));
```

### Use of Sizeof On a Pointer Type\Path 21:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3394 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 1607 | 1607 |
| Object | sizeof | sizeof |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c
Method static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1607.          names = (unsigned char**)cJSON_malloc(numentries *
sizeof(unsigned char*));
```

**Use of Sizeof On a Pointer Type\Path 22:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3395](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3395) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 1613 | 1613 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p) |

```
....
1613.          memset(entries, '\0', sizeof(unsigned char*) *
numentries);
```

**Use of Sizeof On a Pointer Type\Path 23:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3396](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3396) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 1614 | 1614 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p) |

```
....
1614.          memset(names, '\0', sizeof(unsigned char*) * numentries);
```

### Use of Sizeof On a Pointer Type\Path 24:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3397 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c |
| Line | 98 | 98 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c |
| Method | int init_ds_data(ds_partition_t *partition) |

```
....
98.   partition->data = (ds_data_t**)shm_malloc( sizeof(ds_data_t*) );
```

### Use of Sizeof On a Pointer Type\Path 25:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3398 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27596-TP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27596-TP.c |
| Line | 84 | 84 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27596-TP.c |
| Method | static int create_codec_lumps(struct sip_msg * msg) |

```
....
84.   memset(lumps, 0, MAX_STREAMS * sizeof(struct lump*));
```

### Use of Sizeof On a Pointer Type\Path 26:

| | |
|---|---|
| Severity | Low |

| | |
|---|---|
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3399 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27600-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27600-FP.c |
| Line | 84 | 84 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27600-FP.c |
| Method | static int create_codec_lumps(struct sip_msg * msg) |

```
....
84.    memset(lumps, 0, MAX_STREAMS * sizeof(struct lump*));
```

### Use of Sizeof On a Pointer Type\Path 27:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3400 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27601-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27601-FP.c |
| Line | 84 | 84 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-27601-FP.c |
| Method | static int create_codec_lumps(struct sip_msg * msg) |

```
....
84.    memset(lumps, 0, MAX_STREAMS * sizeof(struct lump*));
```

### Use of Sizeof On a Pointer Type\Path 28:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3401 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 1311 | 1311 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Method | static unsigned char *print_array(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p) |

```
....
1311.           entries = (unsigned char**)cJSON_malloc(numentries *
sizeof(unsigned char*));
```

**Use of Sizeof On a Pointer Type\Path 29:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3402 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 1316 | 1316 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Method | static unsigned char *print_array(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p) |

```
....
1316.           memset(entries, '\0', numentries * sizeof(unsigned
char*));
```

**Use of Sizeof On a Pointer Type\Path 30:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3403 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE- | OpenSIPS@@opensips-3.1.0-beta-CVE- |

| | | |
|---|---|---|
| | 2023-28096-FP.c | 2023-28096-FP.c |
| Line | 1651 | 1651 |
| Object | sizeof | sizeof |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c
Method static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1651.           entries = (unsigned char**)cJSON_malloc(numentries *
sizeof(unsigned char*));
```

### Use of Sizeof On a Pointer Type\Path 31:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3404 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 1656 | 1656 |
| Object | sizeof | sizeof |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c
Method static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1656.           names = (unsigned char**)cJSON_malloc(numentries *
sizeof(unsigned char*));
```

### Use of Sizeof On a Pointer Type\Path 32:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3405 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 1662 | 1662 |

| | | |
|---|---|---|
| Object | sizeof | sizeof |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c
Method static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1662.         memset(entries, '\0', sizeof(unsigned char*) *
numentries);
```

**Use of Sizeof On a Pointer Type\Path 33:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3406](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3406) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 1663 | 1663 |
| Object | sizeof | sizeof |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c
Method static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1663.         memset(names, '\0', sizeof(unsigned char*) * numentries);
```

**Use of Sizeof On a Pointer Type\Path 34:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3407](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3407) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c |
| Line | 80 | 80 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c |
| Method | int init_ds_data(ds_partition_t *partition) |

```
....
80.    partition->data = (ds_data_t**)shm_malloc( sizeof(ds_data_t*) );
```

### Use of Sizeof On a Pointer Type\Path 35:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3408 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27596-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27596-TP.c |
| Line | 84 | 84 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.1-CVE-2023-27596-TP.c |
| Method | static int create_codec_lumps(struct sip_msg * msg) |

```
....
84.    memset(lumps, 0, MAX_STREAMS * sizeof(struct lump*));
```

### Use of Sizeof On a Pointer Type\Path 36:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3409 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27600-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27600-TP.c |
| Line | 84 | 84 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.1-CVE-2023-27600-TP.c |
| Method | static int create_codec_lumps(struct sip_msg * msg) |

```
....
84.    memset(lumps, 0, MAX_STREAMS * sizeof(struct lump*));
```

### Use of Sizeof On a Pointer Type\Path 37:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3410 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-27601-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-27601-TP.c |
| Line | 84 | 84 |
| Object | sizeof | sizeof |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-27601-TP.c
Method static int create_codec_lumps(struct sip_msg * msg)

```
....
84.    memset(lumps, 0, MAX_STREAMS * sizeof(struct lump*));
```

### Use of Sizeof On a Pointer Type\Path 38:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3411 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 1312 | 1312 |
| Object | sizeof | sizeof |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c
Method static unsigned char *print_array(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1312.          entries = (unsigned char**)cJSON_malloc(numentries *
sizeof(unsigned char*));
```

### Use of Sizeof On a Pointer Type\Path 39:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20 |

| | |
|---|---|
| | 043&pathid=3412 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 1317 | 1317 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_array(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p) |

```
....
1317.          memset(entries, '\0', numentries * sizeof(unsigned
char*));
```

### Use of Sizeof On a Pointer Type\Path 40:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3413 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 1652 | 1652 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p) |

```
....
1652.          entries = (unsigned char**)cJSON_malloc(numentries *
sizeof(unsigned char*));
```

### Use of Sizeof On a Pointer Type\Path 41:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3414 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 1657 | 1657 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p) |

```
....
1657.           names = (unsigned char**)cJSON_malloc(numentries *
sizeof(unsigned char*));
```

**Use of Sizeof On a Pointer Type\Path 42:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3415 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 1663 | 1663 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p) |

```
....
1663.           memset(entries, '\0', sizeof(unsigned char*) *
numentries);
```

**Use of Sizeof On a Pointer Type\Path 43:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3416 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023- | OpenSIPS@@opensips-3.1.1-CVE-2023- |

| | | |
|---|---|---|
| | 28096-TP.c | 28096-TP.c |
| Line | 1664 | 1664 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p) |

```
....
1664.          memset(names, '\0', sizeof(unsigned char*) * numentries);
```

### Use of Sizeof On a Pointer Type\Path 44:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3417 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-28099-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28099-TP.c |
| Line | 80 | 80 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.1-CVE-2023-28099-TP.c |
| Method | int init_ds_data(ds_partition_t *partition) |

```
....
80.    partition->data = (ds_data_t**)shm_malloc( sizeof(ds_data_t*) );
```

### Use of Sizeof On a Pointer Type\Path 45:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3418 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.2-CVE-2023-27596-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-27596-TP.c |
| Line | 84 | 84 |
| Object | sizeof | sizeof |

Code Snippet
File Name OpenSIPS@@opensips-3.1.2-CVE-2023-27596-TP.c
Method static int create_codec_lumps(struct sip_msg * msg)

```
....
84.    memset(lumps, 0, MAX_STREAMS * sizeof(struct lump*));
```

### Use of Sizeof On a Pointer Type\Path 46:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3419 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.2-CVE-2023-27600-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-27600-TP.c |
| Line | 84 | 84 |
| Object | sizeof | sizeof |

Code Snippet
File Name OpenSIPS@@opensips-3.1.2-CVE-2023-27600-TP.c
Method static int create_codec_lumps(struct sip_msg * msg)

```
....
84.    memset(lumps, 0, MAX_STREAMS * sizeof(struct lump*));
```

### Use of Sizeof On a Pointer Type\Path 47:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3420 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.2-CVE-2023-27601-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-27601-TP.c |
| Line | 84 | 84 |
| Object | sizeof | sizeof |

Code Snippet
File Name OpenSIPS@@opensips-3.1.2-CVE-2023-27601-TP.c
Method static int create_codec_lumps(struct sip_msg * msg)

```
....
84.    memset(lumps, 0, MAX_STREAMS * sizeof(struct lump*));
```

### Use of Sizeof On a Pointer Type\Path 48:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3421 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 1312 | 1312 |
| Object | sizeof | sizeof |

Code Snippet

File Name OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c

Method static unsigned char *print_array(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1312.          entries = (unsigned char**)cJSON_malloc(numentries *
sizeof(unsigned char*));
```

### Use of Sizeof On a Pointer Type\Path 49:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3422 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 1317 | 1317 |
| Object | sizeof | sizeof |

Code Snippet

File Name OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c

Method static unsigned char *print_array(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1317.          memset(entries, '\0', numentries * sizeof(unsigned
char*));
```

### Use of Sizeof On a Pointer Type\Path 50:

| | |
|---|---|
| Severity | Low |

| | |
|---|---|
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3423 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 1652 | 1652 |
| Object | sizeof | sizeof |

Code Snippet

File Name OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c

Method static unsigned char *print_object(const cJSON *item, size_t depth, cjbool fmt, printbuffer *p)

```
....
1652.           entries = (unsigned char**)cJSON_malloc(numentries * 
sizeof(unsigned char*));
```

## Unchecked Return Value

Query Path:

CPP\Cx\CPP Low Visibility\Unchecked Return Value Version:1

### Categories

NIST SP 800-53: SI-11 Error Handling (P2)

*Description*

**Unchecked Return Value\Path 1:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3309 |
| Status | New |

The httpAddrString method calls the snprintf function, at line 503 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-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 | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 537 | 537 |
| Object | snprintf | snprintf |

Code Snippet

File Name OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c

Method httpAddrString(const http_addr_t *addr,       /* I - Address to convert */

```
....
537.        snprintf(s, (size_t)slen, "%d.%d.%d.%d", (temp >> 24) & 255,
```

**Unchecked Return Value\Path 2:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3310](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3310) |
| Status | New |

The httpAddrString method calls the snprintf function, at line 503 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-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 | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 640 | 640 |
| Object | snprintf | snprintf |

Code Snippet
File Name OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c
Method httpAddrString(const http_addr_t *addr, /* I - Address to convert */

```
....
640.        snprintf(s, (size_t)slen, "[v1.%s]", temps);
```

**Unchecked Return Value\Path 3:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3311](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3311) |
| Status | New |

The httpGetHostname method calls the snprintf function, at line 796 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-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 | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 848 | 848 |
| Object | snprintf | snprintf |

Code Snippet
File Name OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c

| | |
|---|---|
| Method | httpGetHostname(http_t *http,          /* I - HTTP connection or NULL */ |

```
....
848.            snprintf(s, (size_t)slen, "%s.local.", localStr);
```

**Unchecked Return Value\Path 4:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3312 |
| Status | New |

The httpAddrString method calls the snprintf function, at line 503 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-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 | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 537 | 537 |
| Object | snprintf | snprintf |

Code Snippet

| | |
|---|---|
| File Name | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Method | httpAddrString(const http_addr_t *addr,       /* I - Address to convert */ |

```
....
537.        snprintf(s, (size_t)slen, "%d.%d.%d.%d", (temp >> 24) & 255,
```

**Unchecked Return Value\Path 5:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3313 |
| Status | New |

The httpAddrString method calls the snprintf function, at line 503 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-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 | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 640 | 640 |
| Object | snprintf | snprintf |

Code Snippet

| | |
|---|---|
| File Name | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Method | httpAddrString(const http_addr_t *addr,        /* I - Address to convert */ |

```
....
640.        snprintf(s, (size_t)slen, "[v1.%s]", temps);
```

### Unchecked Return Value\Path 6:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3314](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3314) |
| Status | New |

The httpGetHostname method calls the snprintf function, at line 794 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-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 | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 846 | 846 |
| Object | snprintf | snprintf |

Code Snippet

| | |
|---|---|
| File Name | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Method | httpGetHostname(http_t *http,            /* I - HTTP connection or NULL */ |

```
....
846.            snprintf(s, (size_t)slen, "%s.local.", localStr);
```

### Unchecked Return Value\Path 7:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3315](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3315) |
| Status | New |

The httpAddrString method calls the snprintf function, at line 503 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-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 | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 537 | 537 |
| Object | snprintf | snprintf |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpAddrString(const http_addr_t *addr, /* I - Address to convert */

```
....
537.        snprintf(s, (size_t)slen, "%d.%d.%d.%d", (temp >> 24) & 255,
```

**Unchecked Return Value\Path 8:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3316 |
| Status | New |

The httpAddrString method calls the snprintf function, at line 503 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-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 | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 640 | 640 |
| Object | snprintf | snprintf |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpAddrString(const http_addr_t *addr, /* I - Address to convert */

```
....
640.        snprintf(s, (size_t)slen, "[v1.%s]", temps);
```

**Unchecked Return Value\Path 9:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3317 |
| Status | New |

The httpGetHostname method calls the snprintf function, at line 794 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-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 | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 846 | 846 |
| Object | snprintf | snprintf |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpGetHostname(http_t *http, /* I - HTTP connection or NULL */

```
....
846.            snprintf(s, (size_t)slen, "%s.local.", localStr);
```

**Unchecked Return Value\Path 10:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3318 |
| Status | New |

The httpAddrString method calls the snprintf function, at line 503 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-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 | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 537 | 537 |
| Object | snprintf | snprintf |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpAddrString(const http_addr_t *addr, /* I - Address to convert */

```
....
537.        snprintf(s, (size_t)slen, "%d.%d.%d.%d", (temp >> 24) & 255,
```

**Unchecked Return Value\Path 11:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3319 |
| Status | New |

The httpAddrString method calls the snprintf function, at line 503 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-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 | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 640 | 640 |

| | | |
|---|---|---|
| Object | snprintf | snprintf |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpAddrString(const http_addr_t *addr, /* I - Address to convert */

```
....
640.       snprintf(s, (size_t)slen, "[v1.%s]", temps);
```

**Unchecked Return Value\Path 12:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3320 |
| Status | New |

The httpGetHostname method calls the snprintf function, at line 794 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-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 | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 846 | 846 |
| Object | snprintf | snprintf |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpGetHostname(http_t *http, /* I - HTTP connection or NULL */

```
....
846.           snprintf(s, (size_t)slen, "%s.local.", localStr);
```

**Unchecked Return Value\Path 13:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3321 |
| Status | New |

The myeid_get_info method calls the snprintf function, at line 1725 of OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

| | Source | Destination |
|---|---|---|
| File | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c |

| | | |
|---|---|---|
| Line | 1753 | 1753 |
| Object | snprintf | snprintf |

Code Snippet

| | |
|---|---|
| File Name | OpenSC@@OpenSC-0.23.0-rc1-CVE-2023-4535-FP.c |
| Method | static int myeid_get_info(struct sc_card *card, u8 *rbuf, size_t buflen) |

```
....
1753.        snprintf(card_name_buf, sizeof(card_name_buf),
```

**Unchecked Return Value\Path 14:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3322 |
| Status | New |

The cJSON_Version method calls the sprintf function, at line 90 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-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 | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 93 | 93 |
| Object | sprintf | sprintf |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Method | extern const char* cJSON_Version(void) |

```
....
93.        sprintf(version, "%i.%i.%i", CJSON_VERSION_MAJOR,
CJSON_VERSION_MINOR, CJSON_VERSION_PATCH);
```

**Unchecked Return Value\Path 15:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3323 |
| Status | New |

The *print_number method calls the sprintf function, at line 358 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-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 | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 392 | 392 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c
Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
392.                    sprintf((char*)str, "%d", item->valueint);
```

**Unchecked Return Value\Path 16:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3324](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3324) |
| Status | New |

The *print_number method calls the sprintf function, at line 358 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-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 | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 413 | 413 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c
Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
413.                         sprintf((char*)str, "null");
```

**Unchecked Return Value\Path 17:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3325](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3325) |
| Status | New |

The *print_number method calls the sprintf function, at line 358 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-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 | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 417 | 417 |
| Object | sprintf | sprintf |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_number(const cJSON *item, printbuffer *p) |

```
....
417.                    sprintf((char*)str, "%.0f", d);
```

### Unchecked Return Value\Path 18:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3326 |
| Status | New |

The *print_number method calls the sprintf function, at line 358 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-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 | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 421 | 421 |
| Object | sprintf | sprintf |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_number(const cJSON *item, printbuffer *p) |

```
....
421.                    sprintf((char*)str, "%e", d);
```

### Unchecked Return Value\Path 19:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3327 |
| Status | New |

The *print_number method calls the sprintf function, at line 358 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-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 | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 425 | 425 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c
Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
425.                    sprintf((char*)str, "%f", d);
```

**Unchecked Return Value\Path 20:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3328 |
| Status | New |

The *print_string_ptr method calls the sprintf function, at line 670 of OpenSIPS@@opensips-2.4.7-CVE-2023-28096-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 | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c |
| Line | 803 | 803 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
803.                        sprintf((char*)ptr2, "u%04x", token);
```

**Unchecked Return Value\Path 21:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3329 |
| Status | New |

The cJSON_Version method calls the sprintf function, at line 96 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 99 | 99 |
| Object | sprintf | sprintf |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Method | extern const char* cJSON_Version(void) |

```
....
99.      sprintf(version, "%i.%i.%i", CJSON_VERSION_MAJOR,
CJSON_VERSION_MINOR, CJSON_VERSION_PATCH);
```

**Unchecked Return Value\Path 22:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3330 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 415 | 415 |
| Object | sprintf | sprintf |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Method | static unsigned char *print_number(const cJSON *item, printbuffer *p) |

```
....
415.            sprintf((char*)str, "%d", item->valueint);
```

**Unchecked Return Value\Path 23:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3331 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 436 | 436 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c
Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
436.                    sprintf((char*)str, "null");
```

**Unchecked Return Value\Path 24:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3332 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 440 | 440 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c
Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
440.                    sprintf((char*)str, "%.0f", d);
```

**Unchecked Return Value\Path 25:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3333 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 444 | 444 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c
Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
444.                    sprintf((char*)str, "%e", d);
```

**Unchecked Return Value\Path 26:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3334 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 448 | 448 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c
Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
448.                    sprintf((char*)str, "%f", d);
```

**Unchecked Return Value\Path 27:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3335 |

|  |  |
|---|---|
| Status | New |

The *print_string_ptr method calls the sprintf function, at line 693 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c. However, the code does not check the return value from this function, and thus would not detect runtime errors or other unexpected states.

|  | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c |
| Line | 826 | 826 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28096-FP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
826.                            sprintf((char*)ptr2, "u%04x", token);
```

**Unchecked Return Value\Path 28:**

|  |  |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3336 |
| Status | New |

The cJSON_Version method calls the sprintf function, at line 96 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-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 | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 99 | 99 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c
Method extern const char* cJSON_Version(void)

```
....
99.       sprintf(version, "%i.%i.%i", CJSON_VERSION_MAJOR, 
CJSON_VERSION_MINOR, CJSON_VERSION_PATCH);
```

**Unchecked Return Value\Path 29:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3337 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-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 | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 415 | 415 |
| Object | sprintf | sprintf |

Code Snippet

File Name OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c

Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
415.                    sprintf((char*)str, "%d", item->valueint);
```

**Unchecked Return Value\Path 30:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3338 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-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 | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 436 | 436 |
| Object | sprintf | sprintf |

Code Snippet

File Name OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c

Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
436.                        sprintf((char*)str, "null");
```

**Unchecked Return Value\Path 31:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3339 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-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 | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 440 | 440 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c
Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
440.                    sprintf((char*)str, "%.0f", d);
```

**Unchecked Return Value\Path 32:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3340 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-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 | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 444 | 444 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c
Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
444.                    sprintf((char*)str, "%e", d);
```

**Unchecked Return Value\Path 33:**

| | |
|---|---|
| Severity | Low |

| | |
|---|---|
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3341 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-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 | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 448 | 448 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c
Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
448.                        sprintf((char*)str, "%f", d);
```

### Unchecked Return Value\Path 34:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3342 |
| Status | New |

The *print_string_ptr method calls the sprintf function, at line 694 of OpenSIPS@@opensips-3.1.1-CVE-2023-28096-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 | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c |
| Line | 827 | 827 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
827.                            sprintf((char*)ptr2, "u%04x", token);
```

### Unchecked Return Value\Path 35:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3343 |
| Status | New |

The cJSON_Version method calls the sprintf function, at line 96 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-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 | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 99 | 99 |
| Object | sprintf | sprintf |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Method | extern const char* cJSON_Version(void) |

```
....
99.       sprintf(version, "%i.%i.%i", CJSON_VERSION_MAJOR, 
CJSON_VERSION_MINOR, CJSON_VERSION_PATCH);
```

**Unchecked Return Value\Path 36:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3344 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-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 | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 415 | 415 |
| Object | sprintf | sprintf |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_number(const cJSON *item, printbuffer *p) |

```
....
415.                sprintf((char*)str, "%d", item->valueint);
```

### Unchecked Return Value\Path 37:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3345 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-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 | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 436 | 436 |
| Object | sprintf | sprintf |

Code Snippet

File Name OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c

Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
436.                    sprintf((char*)str, "null");
```

### Unchecked Return Value\Path 38:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3346 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-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 | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 440 | 440 |
| Object | sprintf | sprintf |

Code Snippet

File Name OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c

Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
440.                    sprintf((char*)str, "%.0f", d);
```

### Unchecked Return Value\Path 39:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3347](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3347) |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-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 | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 444 | 444 |
| Object | sprintf | sprintf |

| | |
|---|---|
| Code Snippet | |
| File Name | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_number(const cJSON *item, printbuffer *p) |

```
....
444.                    sprintf((char*)str, "%e", d);
```

### Unchecked Return Value\Path 40:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3348](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3348) |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-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 | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 448 | 448 |
| Object | sprintf | sprintf |

| | |
|---|---|
| Code Snippet | |
| File Name | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |

Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
448.                    sprintf((char*)str, "%f", d);
```

### Unchecked Return Value\Path 41:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3349 |
| Status | New |

The *print_string_ptr method calls the sprintf function, at line 694 of OpenSIPS@@opensips-3.1.2-CVE-2023-28096-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 | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c |
| Line | 827 | 827 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.1.2-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
827.                        sprintf((char*)ptr2, "u%04x", token);
```

### Unchecked Return Value\Path 42:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3350 |
| Status | New |

The cJSON_Version method calls the sprintf function, at line 96 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-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 | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Line | 99 | 99 |
| Object | sprintf | sprintf |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Method | extern const char* cJSON_Version(void) |

```
....
99.       sprintf(version, "%i.%i.%i", CJSON_VERSION_MAJOR,
CJSON_VERSION_MINOR, CJSON_VERSION_PATCH);
```

### Unchecked Return Value\Path 43:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3351 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-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 | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Line | 415 | 415 |
| Object | sprintf | sprintf |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_number(const cJSON *item, printbuffer *p) |

```
....
415.                  sprintf((char*)str, "%d", item->valueint);
```

### Unchecked Return Value\Path 44:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3352 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-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 | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Line | 436 | 436 |
| Object | sprintf | sprintf |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_number(const cJSON *item, printbuffer *p) |

```
....
436.                    sprintf((char*)str, "null");
```

**Unchecked Return Value\Path 45:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3353](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3353) |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-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 | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Line | 440 | 440 |
| Object | sprintf | sprintf |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Method | static unsigned char *print_number(const cJSON *item, printbuffer *p) |

```
....
440.                    sprintf((char*)str, "%.0f", d);
```

**Unchecked Return Value\Path 46:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3354](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3354) |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-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 | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Line | 444 | 444 |

| | | |
|---|---|---|
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c
Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
444.                    sprintf((char*)str, "%e", d);
```

**Unchecked Return Value\Path 47:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3355 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-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 | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |
| Line | 448 | 448 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c
Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
448.                    sprintf((char*)str, "%f", d);
```

**Unchecked Return Value\Path 48:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3356 |
| Status | New |

The *print_string_ptr method calls the sprintf function, at line 694 of OpenSIPS@@opensips-3.2.1-CVE-2023-28096-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 | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c |

| | | |
|---|---|---|
| Line | 827 | 827 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.2.1-CVE-2023-28096-TP.c
Method static unsigned char *print_string_ptr(const unsigned char *str, printbuffer *p)

```
....
827.                        sprintf((char*)ptr2, "u%04x", token);
```

**Unchecked Return Value\Path 49:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3357 |
| Status | New |

The cJSON_Version method calls the sprintf function, at line 96 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-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 | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c |
| Line | 99 | 99 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c
Method extern const char* cJSON_Version(void)

```
....
99.         sprintf(version, "%i.%i.%i", CJSON_VERSION_MAJOR, 
CJSON_VERSION_MINOR, CJSON_VERSION_PATCH);
```

**Unchecked Return Value\Path 50:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3358 |
| Status | New |

The *print_number method calls the sprintf function, at line 381 of OpenSIPS@@opensips-3.2.4-CVE-2023-28096-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 | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c | OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c |
| Line | 415 | 415 |
| Object | sprintf | sprintf |

Code Snippet
File Name OpenSIPS@@opensips-3.2.4-CVE-2023-28096-TP.c
Method static unsigned char *print_number(const cJSON *item, printbuffer *p)

```
....
415.                sprintf((char*)str, "%d", item->valueint);
```

## Reliance on DNS Lookups in a Decision

Query Path:
CPP\Cx\CPP Low Visibility\Reliance on DNS Lookups in a Decision Version:0

### Categories

FISMA 2014: Identification And Authentication
NIST SP 800-53: SC-23 Session Authenticity (P1)

*Description*

**Reliance on DNS Lookups in a Decision\Path 1:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=932 |
| Status | New |

The httpAddrLookup method performs a reverse DNS lookup with getnameinfo, at line 321 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c. The application then makes a security decision, error, in OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c line 321, even though this hostname is not reliable and can be easily spoofed.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 393 | 397 |
| Object | getnameinfo | error |

Code Snippet
File Name OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c
Method httpAddrLookup(

```
....
393.        int error = getnameinfo(&addr->addr,
(socklen_t)httpAddrLength(addr), name, (socklen_t)namelen, NULL, 0, 0);
....
397.          if (error == EAI_FAIL)
```

### Reliance on DNS Lookups in a Decision\Path 2:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=933 |
| Status | New |

The httpAddrLookup method performs a reverse DNS lookup with getnameinfo, at line 321 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c. The application then makes a security decision, ==, in OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c line 321, even though this hostname is not reliable and can be easily spoofed.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 393 | 397 |
| Object | getnameinfo | == |

Code Snippet

| | |
|---|---|
| File Name | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Method | httpAddrLookup( |

```
....
393.      int error = getnameinfo(&addr->addr,
(socklen_t)httpAddrLength(addr), name, (socklen_t)namelen, NULL, 0, 0);
....
397.        if (error == EAI_FAIL)
```

### Reliance on DNS Lookups in a Decision\Path 3:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=934 |
| Status | New |

The httpAddrLookup method performs a reverse DNS lookup with getnameinfo, at line 321 of OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c. The application then makes a security decision, error, in OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c line 321, even though this hostname is not reliable and can be easily spoofed.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 393 | 395 |
| Object | getnameinfo | error |

Code Snippet

| | |
|---|---|
| File Name | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |

| | |
|---|---|
| Method | httpAddrLookup( |

```
....
393.        int error = getnameinfo(&addr->addr,
(socklen_t)httpAddrLength(addr), name, (socklen_t)namelen, NULL, 0, 0);
....
395.        if (error)
```

### Reliance on DNS Lookups in a Decision\Path 4:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=935 |
| Status | New |

The httpAddrLookup method performs a reverse DNS lookup with getnameinfo, at line 321 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c. The application then makes a security decision, error, in OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c line 321, even though this hostname is not reliable and can be easily spoofed.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 393 | 397 |
| Object | getnameinfo | error |

| | |
|---|---|
| Code Snippet | |
| File Name | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Method | httpAddrLookup( |

```
....
393.        int error = getnameinfo(&addr->addr,
(socklen_t)httpAddrLength(addr), name, (socklen_t)namelen, NULL, 0, 0);
....
397.          if (error == EAI_FAIL)
```

### Reliance on DNS Lookups in a Decision\Path 5:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=936 |
| Status | New |

The httpAddrLookup method performs a reverse DNS lookup with getnameinfo, at line 321 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c. The application then makes a security decision, ==, in OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c line 321, even though this hostname is not reliable and can be easily spoofed.

| | Source | Destination |
|---|---|---|

| | | |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 393 | 397 |
| Object | getnameinfo | == |

Code Snippet
File Name OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c
Method httpAddrLookup(

```
....
393.       int error = getnameinfo(&addr->addr,
(socklen_t)httpAddrLength(addr), name, (socklen_t)namelen, NULL, 0, 0);
....
397.         if (error == EAI_FAIL)
```

**Reliance on DNS Lookups in a Decision\Path 6:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=937](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=937) |
| Status | New |

The httpAddrLookup method performs a reverse DNS lookup with getnameinfo, at line 321 of OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c. The application then makes a security decision, error, in OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c line 321, even though this hostname is not reliable and can be easily spoofed.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 393 | 395 |
| Object | getnameinfo | error |

Code Snippet
File Name OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c
Method httpAddrLookup(

```
....
393.       int error = getnameinfo(&addr->addr,
(socklen_t)httpAddrLength(addr), name, (socklen_t)namelen, NULL, 0, 0);
....
395.       if (error)
```

**Reliance on DNS Lookups in a Decision\Path 7:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=938](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=938) |

| | |
|---|---|
| Status | New |

The httpAddrLookup method performs a reverse DNS lookup with getnameinfo, at line 321 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c. The application then makes a security decision, error, in OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c line 321, even though this hostname is not reliable and can be easily spoofed.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 393 | 397 |
| Object | getnameinfo | error |

Code Snippet

| | |
|---|---|
| File Name | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Method | httpAddrLookup( |

```
....
393.        int error = getnameinfo(&addr->addr,
(socklen_t)httpAddrLength(addr), name, (socklen_t)namelen, NULL, 0, 0);
....
397.          if (error == EAI_FAIL)
```

### Reliance on DNS Lookups in a Decision\Path 8:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=939 |
| Status | New |

The httpAddrLookup method performs a reverse DNS lookup with getnameinfo, at line 321 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c. The application then makes a security decision, ==, in OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c line 321, even though this hostname is not reliable and can be easily spoofed.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 393 | 397 |
| Object | getnameinfo | == |

Code Snippet

| | |
|---|---|
| File Name | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Method | httpAddrLookup( |

```
....
393.        int error = getnameinfo(&addr->addr,
(socklen_t)httpAddrLength(addr), name, (socklen_t)namelen, NULL, 0, 0);
....
397.          if (error == EAI_FAIL)
```

### Reliance on DNS Lookups in a Decision\Path 9:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=940 |
| Status | New |

The httpAddrLookup method performs a reverse DNS lookup with getnameinfo, at line 321 of OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c. The application then makes a security decision, error, in OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c line 321, even though this hostname is not reliable and can be easily spoofed.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 393 | 395 |
| Object | getnameinfo | error |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpAddrLookup(

```
....
393.        int error = getnameinfo(&addr->addr,
(socklen_t)httpAddrLength(addr), name, (socklen_t)namelen, NULL, 0, 0);
....
395.        if (error)
```

### Reliance on DNS Lookups in a Decision\Path 10:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=941 |
| Status | New |

The httpAddrLookup method performs a reverse DNS lookup with getnameinfo, at line 321 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c. The application then makes a security decision, error, in OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c line 321, even though this hostname is not reliable and can be easily spoofed.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024- | OpenPrinting@@cups-v2.4.8-CVE-2024- |

| | | |
|---|---|---|
| | 35235-TP.c | 35235-TP.c |
| Line | 393 | 397 |
| Object | getnameinfo | error |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpAddrLookup(

```
....
393.       int error = getnameinfo(&addr->addr,
(socklen_t)httpAddrLength(addr), name, (socklen_t)namelen, NULL, 0, 0);
....
397.         if (error == EAI_FAIL)
```

### Reliance on DNS Lookups in a Decision\Path 11:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=942 |
| Status | New |

The httpAddrLookup method performs a reverse DNS lookup with getnameinfo, at line 321 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c. The application then makes a security decision, ==, in OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c line 321, even though this hostname is not reliable and can be easily spoofed.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 393 | 397 |
| Object | getnameinfo | == |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpAddrLookup(

```
....
393.       int error = getnameinfo(&addr->addr,
(socklen_t)httpAddrLength(addr), name, (socklen_t)namelen, NULL, 0, 0);
....
397.         if (error == EAI_FAIL)
```

### Reliance on DNS Lookups in a Decision\Path 12:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=943 |
| Status | New |

The httpAddrLookup method performs a reverse DNS lookup with getnameinfo, at line 321 of OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c. The application then makes a security decision, error, in OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c line 321, even though this hostname is not reliable and can be easily spoofed.

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 393 | 395 |
| Object | getnameinfo | error |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpAddrLookup(

```
....
393.       int error = getnameinfo(&addr->addr,
(socklen_t)httpAddrLength(addr), name, (socklen_t)namelen, NULL, 0, 0);
....
395.       if (error)
```

## Use of Insufficiently Random Values

Query Path:
CPP\Cx\CPP Low Visibility\Use of Insufficiently Random Values Version:0

### Categories

FISMA 2014: Media Protection
NIST SP 800-53: SC-28 Protection of Information at Rest (P1)
OWASP Top 10 2017: A3-Sensitive Data Exposure

*Description*

**Use of Insufficiently Random Values\Path 1:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3303 |
| Status | New |

Method ds_select_dst at line 1477 of OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c uses a weak method rand to produce random values. These values might be used for secret values, personal identifiers or cryptographic input, allowing an attacker to guess the value.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c | OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c |
| Line | 1611 | 1611 |
| Object | rand | rand |

Code Snippet

File Name OpenSIPS@@opensips-2.4.7-CVE-2023-28099-TP.c
Method int ds_select_dst(struct sip_msg *msg, ds_select_ctl_p ds_select_ctl,

```
....
1611.                      ds_hash = rand();
```

### Use of Insufficiently Random Values\Path 2:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3304 |
| Status | New |

Method ds_select_dst at line 1595 of OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c uses a weak method rand to produce random values. These values might be used for secret values, personal identifiers or cryptographic input, allowing an attacker to guess the value.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c | OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c |
| Line | 1721 | 1721 |
| Object | rand | rand |

Code Snippet
File Name OpenSIPS@@opensips-3.1.0-beta-CVE-2023-28099-FP.c
Method int ds_select_dst(struct sip_msg *msg, ds_select_ctl_p ds_select_ctl,

```
....
1721.                      ds_hash = rand();
```

### Use of Insufficiently Random Values\Path 3:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3305 |
| Status | New |

Method ds_select_dst at line 1596 of OpenSIPS@@opensips-3.1.1-CVE-2023-28099-TP.c uses a weak method rand to produce random values. These values might be used for secret values, personal identifiers or cryptographic input, allowing an attacker to guess the value.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.1-CVE-2023-28099-TP.c | OpenSIPS@@opensips-3.1.1-CVE-2023-28099-TP.c |
| Line | 1722 | 1722 |
| Object | rand | rand |

Code Snippet
File Name OpenSIPS@@opensips-3.1.1-CVE-2023-28099-TP.c

| | |
|---|---|
| Method | int ds_select_dst(struct sip_msg *msg, ds_select_ctl_p ds_select_ctl, |

```
....
1722.                    ds_hash = rand();
```

### Use of Insufficiently Random Values\Path 4:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3306](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3306) |
| Status | New |

Method ds_select_dst at line 1596 of OpenSIPS@@opensips-3.1.2-CVE-2023-28099-TP.c uses a weak method rand to produce random values. These values might be used for secret values, personal identifiers or cryptographic input, allowing an attacker to guess the value.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.1.2-CVE-2023-28099-TP.c | OpenSIPS@@opensips-3.1.2-CVE-2023-28099-TP.c |
| Line | 1722 | 1722 |
| Object | rand | rand |

| | |
|---|---|
| Code Snippet | |
| File Name | OpenSIPS@@opensips-3.1.2-CVE-2023-28099-TP.c |
| Method | int ds_select_dst(struct sip_msg *msg, ds_select_ctl_p ds_select_ctl, |

```
....
1722.                    ds_hash = rand();
```

### Use of Insufficiently Random Values\Path 5:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3307](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3307) |
| Status | New |

Method ds_select_dst at line 1644 of OpenSIPS@@opensips-3.2.1-CVE-2023-28099-TP.c uses a weak method rand to produce random values. These values might be used for secret values, personal identifiers or cryptographic input, allowing an attacker to guess the value.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.1-CVE-2023-28099-TP.c | OpenSIPS@@opensips-3.2.1-CVE-2023-28099-TP.c |
| Line | 1770 | 1770 |
| Object | rand | rand |

| | |
|---|---|
| Code Snippet | |
| File Name | OpenSIPS@@opensips-3.2.1-CVE-2023-28099-TP.c |
| Method | int ds_select_dst(struct sip_msg *msg, ds_select_ctl_p ds_select_ctl, |

```
....
1770.                    ds_hash = rand();
```

**Use of Insufficiently Random Values\Path 6:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3308 |
| Status | New |

Method ds_select_dst at line 1644 of OpenSIPS@@opensips-3.2.4-CVE-2023-28099-TP.c uses a weak method rand to produce random values. These values might be used for secret values, personal identifiers or cryptographic input, allowing an attacker to guess the value.

| | Source | Destination |
|---|---|---|
| File | OpenSIPS@@opensips-3.2.4-CVE-2023-28099-TP.c | OpenSIPS@@opensips-3.2.4-CVE-2023-28099-TP.c |
| Line | 1770 | 1770 |
| Object | rand | rand |

Code Snippet

| | |
|---|---|
| File Name | OpenSIPS@@opensips-3.2.4-CVE-2023-28099-TP.c |
| Method | int ds_select_dst(struct sip_msg *msg, ds_select_ctl_p ds_select_ctl, |

```
....
1770.                    ds_hash = rand();
```

## Incorrect Permission Assignment For Critical Resources

Query Path:
CPP\Cx\CPP Low Visibility\Incorrect Permission Assignment For Critical Resources Version:1

### Categories

FISMA 2014: Access Control
NIST SP 800-53: AC-3 Access Enforcement (P1)
OWASP Top 10 2017: A2-Broken Authentication

*Description*

**Incorrect Permission Assignment For Critical Resources\Path 1:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3299 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c |
| Line | 229 | 229 |

| | | |
|---|---|---|
| Object | chmod | chmod |

Code Snippet
File Name OpenPrinting@@cups-v2.4.2-CVE-2024-35235-TP.c
Method httpAddrListen(http_addr_t *addr,	/* I - Address to bind to */

```
....
229.		chmod(addr->un.sun_path, 0140777);
```

**Incorrect Permission Assignment For Critical Resources\Path 2:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3300 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c |
| Line | 229 | 229 |
| Object | chmod | chmod |

Code Snippet
File Name OpenPrinting@@cups-v2.4.3-CVE-2024-35235-TP.c
Method httpAddrListen(http_addr_t *addr,	/* I - Address to bind to */

```
....
229.		chmod(addr->un.sun_path, 0140777);
```

**Incorrect Permission Assignment For Critical Resources\Path 3:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3301 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c |
| Line | 229 | 229 |
| Object | chmod | chmod |

Code Snippet
File Name OpenPrinting@@cups-v2.4.7-CVE-2024-35235-TP.c
Method httpAddrListen(http_addr_t *addr,	/* I - Address to bind to */

```
....
229.        chmod(addr->un.sun_path, 0140777);
```

**Incorrect Permission Assignment For Critical Resources\Path 4:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3302](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=3302) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c | OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c |
| Line | 229 | 229 |
| Object | chmod | chmod |

Code Snippet
File Name OpenPrinting@@cups-v2.4.8-CVE-2024-35235-TP.c
Method httpAddrListen(http_addr_t *addr, /* I - Address to bind to */

```
....
229.        chmod(addr->un.sun_path, 0140777);
```

## Inconsistent Implementations

Query Path:
CPP\Cx\CPP Low Visibility\Inconsistent Implementations Version:0

*Description*

**Inconsistent Implementations\Path 1:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=925](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1020050&projectid=20043&pathid=925) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | OpenRC@@openrc-0.43.1-CVE-2021-42341-FP.c | OpenRC@@openrc-0.43.1-CVE-2021-42341-FP.c |
| Line | 349 | 349 |
| Object | getopt_long | getopt_long |

Code Snippet
File Name OpenRC@@openrc-0.43.1-CVE-2021-42341-FP.c
Method int main(int argc, char **argv)

```
....
349.        while ((opt = getopt_long(argc, argv, getoptstring,
```

# Buffer Overflow LongString

## Risk

### What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

## Cause

### How does it happen

Buffer Overflows can manifest in numerous different variations. In it's most basic form, the attack controls a buffer, which is then copied to a smaller buffer without size verification. Because the attacker's source buffer is larger than the program's target buffer, the attacker's data overwrites whatever is next on the stack, allowing the attacker to control program structures.

Alternatively, the vulnerability could be the result of improper bounds checking; exposing internal memory addresses outside of their valid scope; allowing the attacker to control the size of the target buffer; or various other forms.

## General Recommendations

### How to avoid it

- Always perform proper bounds checking before copying buffers or strings.
- Prefer to use safer functions and structures, e.g. safe string classes over `char*`, `strncpy` over `strcpy`, and so on.
- Consistently apply tests for the size of buffers.
- Do not return variable addresses outside the scope of their variables.

## Source Code Examples

### CPP

### Overflowing Buffers

```
const int BUFFER_SIZE = 10;
char buffer[BUFFER_SIZE];

void copyStringToBuffer(char* inputString)
{
    strcpy(buffer, inputString);
}
```

## Checked Buffers

```
const int BUFFER_SIZE = 10;
const int MAX_INPUT_SIZE = 256;
char buffer[BUFFER_SIZE];

void copyStringToBuffer(char* inputString)

{

    if (strnlen(inputString, MAX_INPUT_SIZE) < sizeof(buffer))
    {
        strncpy(buffer, inputString, sizeof(buffer));
    }
}
```

# Buffer Overflow StrcpyStrcat

## Risk

### What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

## Cause

### How does it happen

Buffer Overflows can manifest in numerous different variations. In it's most basic form, the attack controls a buffer, which is then copied to a smaller buffer without size verification. Because the attacker's source buffer is larger than the program's target buffer, the attacker's data overwrites whatever is next on the stack, allowing the attacker to control program structures.

Alternatively, the vulnerability could be the result of improper bounds checking; exposing internal memory addresses outside of their valid scope; allowing the attacker to control the size of the target buffer; or various other forms.

## General Recommendations

### How to avoid it

- Always perform proper bounds checking before copying buffers or strings.
- Prefer to use safer functions and structures, e.g. safe string classes over `char*`, `strncpy` over `strcpy`, and so on.
- Consistently apply tests for the size of buffers.
- Do not return variable addresses outside the scope of their variables.

## Source Code Examples

# Buffer Overflow boundcpy WrongSizeParam

## Risk

### What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

## Cause

### How does it happen

Buffer Overflows can manifest in numerous different variations. In it's most basic form, the attack controls a buffer, which is then copied to a smaller buffer without size verification. Because the attacker's source buffer is larger than the program's target buffer, the attacker's data overwrites whatever is next on the stack, allowing the attacker to control program structures.

Alternatively, the vulnerability could be the result of improper bounds checking; exposing internal memory addresses outside of their valid scope; allowing the attacker to control the size of the target buffer; or various other forms.

## General Recommendations

### How to avoid it

- Always perform proper bounds checking before copying buffers or strings.
- Prefer to use safer functions and structures, e.g. safe string classes over `char*`, `strncpy` over `strcpy`, and so on.
- Consistently apply tests for the size of buffers.
- Do not return variable addresses outside the scope of their variables.

## Source Code Examples

# MemoryFree on StackVariable

## Risk

**What might happen**

Undefined Behavior may result with a crash. Crashes may give an attacker valuable information about the system and the program internals. Furthermore, it may leave unprotected files (e.g memory) that may be exploited.

## Cause

**How does it happen**

Calling free() on a variable that was not dynamically allocated (e.g. malloc) will result with an Undefined Behavior.

## General Recommendations

**How to avoid it**

Use free() only on dynamically allocated variables in order to prevent unexpected behavior from the compiler.

## Source Code Examples

**CPP**

**Bad - Calling free() on a static variable**

```
void clean_up(){
  char temp[256];
  do_something();
  free(tmp);
  return;
}
```

**Good - Calling free() only on variables that were dynamically allocated**

```
void clean_up(){
  char *buff;
  buff = (char*) malloc(1024);
  free(buff);
  return;
}
```

# Wrong Size t Allocation

## Risk

### What might happen

Incorrect allocation of memory may result in unexpected behavior by either overwriting sections of memory with unexpected values. Under certain conditions where both an incorrect allocation of memory and the values being written can be controlled by an attacker, such an issue may result in execution of malicious code.

## Cause

### How does it happen

Some memory allocation functions require a size value to be provided as a parameter. The allocated size should be derived from the provided value, by providing the length value of the intended source, multiplied by the size of that length. Failure to perform the correct arithmetic to obtain the exact size of the value will likely result in the source overflowing its destination.

## General Recommendations

### How to avoid it

- Always perform the correct arithmetic to determine size.
- Specifically for memory allocation, calculate the allocation size from the allocation source:
    - Derive the size value from the length of intended source to determine the amount of units to be processed.
    - Always programmatically consider the size of the each unit and their conversion to memory units - for example, by using sizeof() on the unit's type.
    - Memory allocation should be a multiplication of the amount of units being written, times the size of each unit.

## Source Code Examples

# 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

### 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;
}
```

# 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

```c
int main(int argc, char* argv[])
{
      printf("%s", argv[1]); // Second parameter is not a formattable string

      return 0;
}
```

# Double Free

**Weakness ID:** 415 *(Weakness Variant)* **Status:** Draft

## Description

### Description Summary

The product calls free() twice on the same memory address, potentially leading to modification of unexpected memory locations.

### Extended Description

When a program calls free() twice with the same argument, the program's memory management data structures become corrupted. This corruption can cause the program to crash or, in some circumstances, cause two later calls to malloc() to return the same pointer. If malloc() returns the same value twice and the program later gives the attacker control over the data that is written into this doubly-allocated memory, the program becomes vulnerable to a buffer overflow attack.

## Alternate Terms

**Double-free**

## Time of Introduction

- Architecture and Design
- Implementation

## Applicable Platforms

### Languages

C

C++

## Common Consequences

| Scope | Effect |
|---|---|
| Access Control | Doubly freeing memory may result in a write-what-where condition, allowing an attacker to execute arbitrary code. |

## Likelihood of Exploit

Low to Medium

## Demonstrative Examples

### Example 1

The following code shows a simple example of a double free vulnerability.

*(Bad Code)*

*Example Language:* **C**

```
char* ptr = (char*)malloc (SIZE);
...
if (abrt) {
free(ptr);
}
...
free(ptr);
```

Double free vulnerabilities have two common (and sometimes overlapping) causes:

- Error conditions and other exceptional circumstances
- Confusion over which part of the program is responsible for freeing the memory

Although some double free vulnerabilities are not much more complicated than the previous example, most are spread out across hundreds of lines of code or even different files. Programmers seem particularly susceptible to freeing global variables more than once.

### Example 2

While contrived, this code should be exploitable on Linux distributions which do not ship with heap-chunk check summing turned on.

*(Bad Code)*

*Example Language:* **C**

```
#include <stdio.h>
#include <unistd.h>
#define BUFSIZE1 512
#define BUFSIZE2 ((BUFSIZE1/2) - 8)

int main(int argc, char **argv) {
char *buf1R1;
char *buf2R1;
char *buf1R2;
buf1R1 = (char *) malloc(BUFSIZE2);
buf2R1 = (char *) malloc(BUFSIZE2);
free(buf1R1);
free(buf2R1);
buf1R2 = (char *) malloc(BUFSIZE1);
strncpy(buf1R2, argv[1], BUFSIZE1-1);
free(buf2R1);
free(buf1R2);
}
```

## Observed Examples

| Reference | Description |
|---|---|
| CVE-2004-0642 | Double free resultant from certain error conditions. |
| CVE-2004-0772 | Double free resultant from certain error conditions. |
| CVE-2005-1689 | Double free resultant from certain error conditions. |
| CVE-2003-0545 | Double free from invalid ASN.1 encoding. |
| CVE-2003-1048 | Double free from malformed GIF. |
| CVE-2005-0891 | Double free from malformed GIF. |
| CVE-2002-0059 | Double free from malformed compressed data. |

## Potential Mitigations

#### Phase: Architecture and Design

Choose a language that provides automatic memory management.

#### Phase: Implementation

Ensure that each allocation is freed only once. After freeing a chunk, set the pointer to NULL to ensure the pointer cannot be freed again. In complicated error conditions, be sure that clean-up routines respect the state of allocation properly. If the language is object oriented, ensure that object destructors delete each chunk of memory only once.

#### Phase: Implementation

Use a static analysis tool to find double free instances.

## Relationships

| Nature | Type | ID | Name | View(s) this relationship pertains to |
|---|---|---|---|---|
| ChildOf | Weakness Class | 398 | Indicator of Poor Code Quality | **Seven Pernicious Kingdoms (primary)700** |
| ChildOf | Category | 399 | Resource Management Errors | **Development Concepts (primary)699** |
| ChildOf | Category | 633 | Weaknesses that Affect Memory | **Resource-specific Weaknesses (primary)631** |
| ChildOf | Weakness Base | 666 | Operation on Resource in Wrong Phase of | **Research Concepts (primary)1000** |

| | | | | |
|---|---|---|---|---|
| | | | Lifetime | |
| ChildOf | Weakness Class | 675 | Duplicate Operations on Resource | Research Concepts1000 |
| ChildOf | Category | 742 | CERT C Secure Coding Section 08 - Memory Management (MEM) | **Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734** |
| PeerOf | Weakness Base | 123 | Write-what-where Condition | Research Concepts1000 |
| PeerOf | Weakness Base | 416 | Use After Free | Development Concepts699<br>Research Concepts1000 |
| MemberOf | View | 630 | Weaknesses Examined by SAMATE | **Weaknesses Examined by SAMATE (primary)630** |
| PeerOf | Weakness Base | 364 | Signal Handler Race Condition | Research Concepts1000 |

## Relationship Notes

This is usually resultant from another weakness, such as an unhandled error or race condition between threads. It could also be primary to weaknesses such as buffer overflows.

## Affected Resources

- Memory

## Taxonomy Mappings

| Mapped Taxonomy Name | Node ID | Fit | Mapped Node Name |
|---|---|---|---|
| PLOVER | | | DFREE - Double-Free Vulnerability |
| 7 Pernicious Kingdoms | | | Double Free |
| CLASP | | | Doubly freeing memory |
| CERT C Secure Coding | MEM00-C | | Allocate and free memory in the same module, at the same level of abstraction |
| CERT C Secure Coding | MEM01-C | | Store a new value in pointers immediately after free() |
| CERT C Secure Coding | MEM31-C | | Free dynamically allocated memory exactly once |

## White Box Definitions

A weakness where code path has:

1. start statement that relinquishes a dynamically allocated memory resource
2. end statement that relinquishes the dynamically allocated memory resource

## Maintenance Notes

It could be argued that Double Free would be most appropriately located as a child of "Use after Free", but "Use" and "Release" are considered to be distinct operations within vulnerability theory, therefore this is more accurately "Release of a Resource after Expiration or Release", which doesn't exist yet.

## Content History

**Submissions**

| Submission Date | Submitter | Organization | Source |
|---|---|---|---|
| | PLOVER | | Externally Mined |

**Modifications**

| Modification Date | Modifier | Organization | Source |
|---|---|---|---|
| 2008-07-01 | Eric Dalci | Cigital | External |
| | updated Potential Mitigations, Time of Introduction | | |
| 2008-08-01 | | KDM Analytics | External |
| | added/updated white box definitions | | |
| 2008-09-08 | CWE Content Team | MITRE | Internal |
| | updated Applicable Platforms, Common Consequences, Description, Maintenance Notes, Relationships, Other Notes, Relationship Notes, Taxonomy Mappings | | |
| 2008-11-24 | CWE Content Team | MITRE | Internal |

| | | | |
|---|---|---|---|
| | updated Relationships, Taxonomy Mappings | | |
| 2009-05-27 | CWE Content Team | MITRE | Internal |
| | updated Demonstrative Examples | | |
| 2009-10-29 | CWE Content Team | MITRE | Internal |
| | updated Other Notes | | |

BACK TO TOP

# Use of Hard coded Cryptographic Key

## Risk

### What might happen

Static, unchangeable encryption keys in the source code can be stolen by an attacker with access to the source code or the application binaries. Once the attacker has the encryption key, this can be used to gain access to any encrypted secret data, thus violating the confidentiality of the data. Furthermore, it would be impossible to replace the encryption key once stolen. Note that if this is a product that can be installed numerous times, the encryption key will always be the same, allowing an attacker to break all instances at the same cost.

## Cause

### How does it happen

The application code uses an encryption key to encrypt and decrypt sensitive data. While it is important to create this encryption key randomly and keep it secret, the application has a single, static key embedded in plain text in the source code.

An attacker could gain access to the source code - whether in the source control system, developer workstations, or the server filesystem or product binaries themselves. Once the attacker has gained access to the source code, it is trivial to retrieve the plain text encryption key and use it to decrypt the sensitive data that the application was protecting.

## General Recommendations

### How to avoid it

Generic Guidance:

- Do not store any sensitive information, such as encryption keys, in plain text.
- Never hardcode encryption keys in the application source code.
- Implement proper key management, including dynamically generating random keys, protecting keys, and replacing keys as necessary.

Specific Recommendations:

- Remove the hardcoded encryption key from the application source code. Instead, retrieve the key from an external, protected store.

## Source Code Examples

### Java

**Common example of hardcoded encryption key**

```
//Generate a key
string encryptionKey = "EncryptionKey123"

//Encrypt the data
SecretKeySpec keySpec = new SecretKeySpec(encryptionKey.getBytes(), "AES");
Cipher cipher = Cipher.getInstance("AES/CBC/PKCS7Padding");
cipher.init(Cipher.ENCRYPT_MODE, keySpec);
output = cipher.doFinal(input)
```

# Heap Inspection

## Risk

### What might happen

All variables stored by the application in unencrypted memory can potentially be retrieved by an unauthorized user, with privlieged access to the machine. For example, a privileged attacker could attach a debugger to the running process, or retrieve the process's memory from the swapfile or crash dump file.
Once the attacker finds the user passwords in memory, these can be reused to easily impersonate the user to the system.

## Cause

### How does it happen

String variables are immutable - in other words, once a string variable is assigned, its value cannot be changed or removed. Thus, these strings may remain around in memory, possibly in multiple locations, for an indefinite period of time until the garbage collector happens to remove it. Sensitive data, such as passwords, will remain exposed in memory as plaintext with no control over their lifetime.

## General Recommendations

### How to avoid it

Generic Guidance:

- Do not store senstiive data, such as passwords or encryption keys, in memory in plaintext, even for a short period of time.
- Prefer to use specialized classes that store encrypted memory.
- Alternatively, store secrets temporarily in mutable data types, such as byte arrays, and then promptly zeroize the memory locations.

Specific Recommendations - Java:

- Instead of storing passwords in immutable strings, prefer to use an encrypted memory object, such as SealedObject.

Specific Recommendations - .NET:

- Instead of storing passwords in immutable strings, prefer to use an encrypted memory object, such as SecureString or ProtectedData.

## Source Code Examples

### Java

### Plaintext Password in Immutable String

```
class Heap_Inspection
{
  private string password;

  void setPassword()
```

```
 {
      password = System.console().readLine("Enter your password: ");
  }
}
```

### Password Protected in Memory

```
class Heap_Inspection_Fixed
{
  private SealedObject password;

  void setPassword()
 {
      byte[] sKey = getKeyFromConfig();
      Cipher c = Cipher.getInstance("AES");
      c.init(Cipher.ENCRYPT_MODE, sKey);

      char[] input = System.console().readPassword("Enter your password: ");
      password = new SealedObject(Arrays.asList(input), c);

    //Zero out the possible password, for security.
    Arrays.fill(password, '0');
  }
}
```

### CPP

### Vulnerable C code

```
/* Vulnerable to heap inspection */

#include <stdio.h>


void somefunc(){
      printf("Yea, I'm just being called for the heap of it..\n");
}

void authfunc(){
        char* password = (char *) malloc(256);
        char ch;
        ssize_t k;
            int i=0;
        while(k = read(0, &ch, 1) > 0)
        {
                if (ch == '\n'){
                        password[i]='\0';
                        break;
                } else{
                        password[i++]=ch;
                        fflush(0);
                }
        }
        printf("Password: %s\n",&password[0]);
}

int main()

{

    printf("Please enter a password:\n");

      authfunc();
      printf("You can now dump memory to find this password!");
      somefunc();
      gets();

}
```

### Safe C code

```
/* Pesumably safe heap */

#include <stdio.h>
#include <string.h>

#define STDIN_FILENO 0

void somefunc(){
        printf("Yea, I'm just being called for the heap of it..\n");
}

void authfunc(){
      char* password = (char*) malloc(256);
      int i=0;
      char ch;
      ssize_t k;
      while(k = read(STDIN_FILENO, &ch, 1) > 0)
      {
             if (ch == '\n'){
                   password[i]='\0';
                   break;
             } else{
                   password[i++]=ch;
                   fflush(0);
             }
      }
      i=0;
      memset(password,'\0',256);
}

int main()

{

      printf("Please enter a password:\n");
      authfunc();
      somefunc();
      char ch;
      while(read(STDIN_FILENO, &ch, 1) > 0)
      {
             if (ch == '\n')
                   break;
      }
}
```

## Failure to Release Memory Before Removing Last Reference ('Memory Leak')

**Weakness ID:** 401 *(Weakness Base)* **Status:** Draft

### Description

## Description Summary

The software does not sufficiently track and release allocated memory after it has been used, which slowly consumes remaining memory.

## Extended Description

This is often triggered by improper handling of malformed data or unexpectedly interrupted sessions.

### Terminology Notes

"memory leak" has sometimes been used to describe other kinds of issues, e.g. for information leaks in which the contents of memory are inadvertently leaked (CVE-2003-0400 is one such example of this terminology conflict).

### Time of Introduction

- Architecture and Design
- Implementation

### Applicable Platforms

## Languages

C

C++

### Modes of Introduction

Memory leaks have two common and sometimes overlapping causes:

- Error conditions and other exceptional circumstances
- Confusion over which part of the program is responsible for freeing the memory

### Common Consequences

| Scope | Effect |
|---|---|
| Availability | Most memory leaks result in general software reliability problems, but if an attacker can intentionally trigger a memory leak, the attacker might be able to launch a denial of service attack (by crashing or hanging the program) or take advantage of other unexpected program behavior resulting from a low memory condition. |

### Likelihood of Exploit

Medium

### Demonstrative Examples

## Example 1

The following C function leaks a block of allocated memory if the call to read() fails to return the expected number of bytes:

*(Bad Code)*

*Example Language:* **C**

```
char* getBlock(int fd) {
char* buf = (char*) malloc(BLOCK_SIZE);
if (!buf) {
return NULL;
}
if (read(fd, buf, BLOCK_SIZE) != BLOCK_SIZE) {

return NULL;
}
return buf;
}
```

## Example 2

Here the problem is that every time a connection is made, more memory is allocated. So if one just opened up more and more connections, eventually the machine would run out of memory.

*(Bad Code)*

*Example Language:* **C**

```
bar connection(){
foo = malloc(1024);
return foo;
}
endConnection(bar foo) {

free(foo);
}
int main() {

while(1) //thread 1
//On a connection
foo=connection(); //thread 2
//When the connection ends
endConnection(foo)
}
```

## Observed Examples

| Reference | Description |
|---|---|
| CVE-2005-3119 | Memory leak because function does not free() an element of a data structure. |
| CVE-2004-0427 | Memory leak when counter variable is not decremented. |
| CVE-2002-0574 | Memory leak when counter variable is not decremented. |
| CVE-2005-3181 | Kernel uses wrong function to release a data structure, preventing data from being properly tracked by other code. |
| CVE-2004-0222 | Memory leak via unknown manipulations as part of protocol test suite. |
| CVE-2001-0136 | Memory leak via a series of the same command. |

## Potential Mitigations

Pre-design: Use a language or compiler that performs automatic bounds checking.

**Phase: Architecture and Design**

Use an abstraction library to abstract away risky APIs. Not a complete solution.

Pre-design through Build: The Boehm-Demers-Weiser Garbage Collector or valgrind can be used to detect leaks in code. This is not a complete solution as it is not 100% effective.

## Relationships

| Nature | Type | ID | Name | View(s) this relationship pertains to |
|---|---|---|---|---|
| ChildOf | Weakness Class | 398 | Indicator of Poor Code Quality | **Seven Pernicious Kingdoms (primary)700** |
| ChildOf | Category | 399 | Resource Management Errors | **Development Concepts (primary)699** |
| ChildOf | Category | 633 | Weaknesses that Affect Memory | **Resource-specific Weaknesses (primary)631** |
| ChildOf | Category | 730 | OWASP Top Ten 2004 Category A9 - Denial of Service | **Weaknesses in OWASP Top Ten (2004) (primary)711** |
| ChildOf | Weakness Base | 772 | Missing Release of Resource after Effective | **Research Concepts (primary)1000** |

| | | | | |
|---|---|---|---|---|
| | | | Lifetime | |
| MemberOf | View | 630 | Weaknesses Examined by SAMATE | **Weaknesses Examined by SAMATE (primary)630** |
| CanFollow | Weakness Class | 390 | Detection of Error Condition Without Action | Research Concepts1000 |

## Relationship Notes

This is often a resultant weakness due to improper handling of malformed data or early termination of sessions.

## Affected Resources

- Memory

## Functional Areas

- Memory management

## Taxonomy Mappings

| Mapped Taxonomy Name | Node ID | Fit | Mapped Node Name |
|---|---|---|---|
| PLOVER | | | Memory leak |
| 7 Pernicious Kingdoms | | | Memory Leak |
| CLASP | | | Failure to deallocate data |
| OWASP Top Ten 2004 | A9 | CWE More Specific | Denial of Service |

## White Box Definitions

A weakness where the code path has:

1. start statement that allocates dynamically allocated memory resource

2. end statement that loses identity of the dynamically allocated memory resource creating situation where dynamically allocated memory resource is never relinquished

Where "loses" is defined through the following scenarios:

1. identity of the dynamic allocated memory resource never obtained

2. the statement assigns another value to the data element that stored the identity of the dynamically allocated memory resource and there are no aliases of that data element

3. identity of the dynamic allocated memory resource obtained but never passed on to function for memory resource release

4. the data element that stored the identity of the dynamically allocated resource has reached the end of its scope at the statement and there are no aliases of that data element

## References

J. Whittaker and H. Thompson. "How to Break Software Security". Addison Wesley. 2003.

## Content History

**Submissions**

| Submission Date | Submitter | Organization | Source |
|---|---|---|---|
| | PLOVER | | Externally Mined |

**Modifications**

| Modification Date | Modifier | Organization | Source |
|---|---|---|---|
| 2008-07-01 | Eric Dalci | Cigital | External |
| | updated Time of Introduction | | |
| 2008-08-01 | | KDM Analytics | External |
| | added/updated white box definitions | | |
| 2008-08-15 | | Veracode | External |
| | Suggested OWASP Top Ten 2004 mapping | | |
| 2008-09-08 | CWE Content Team | MITRE | Internal |
| | updated Applicable Platforms, Common Consequences, Relationships, Other Notes, References, Relationship Notes, Taxonomy Mappings, Terminology Notes | | |
| 2008-10-14 | CWE Content Team | MITRE | Internal |
| | updated Description | | |
| 2009-03-10 | CWE Content Team | MITRE | Internal |
| | updated Other Notes | | |
| 2009-05-27 | CWE Content Team | MITRE | Internal |
| | updated Name | | |
| 2009-07-17 | KDM Analytics | | External |
| | Improved the White Box Definition | | |

| | | | |
|---|---|---|---|
| 2009-07-27 | CWE Content Team | MITRE | Internal |
| | updated White Box Definitions | | |
| 2009-10-29 | CWE Content Team | MITRE | Internal |
| | updated Modes of Introduction, Other Notes | | |
| 2010-02-16 | CWE Content Team | MITRE | Internal |
| | updated Relationships | | |

## Previous Entry Names

| Change Date | Previous Entry Name |
|---|---|
| 2008-04-11 | Memory Leak |
| 2009-05-27 | Failure to Release Memory Before Removing Last Reference (aka 'Memory Leak') |

BACK TO TOP

# Use of 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 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());
```

# Wrong Memory Allocation

## Risk

### What might happen

Incorrect allocation of memory may result in unexpected behavior by either overwriting sections of memory with unexpected values. Under certain conditions where both an incorrect allocation of memory and the values being written can be controlled by an attacker, such an issue may result in execution of malicious code.

## Cause

### How does it happen

Some memory allocation functions require a size value to be provided as a parameter. The allocated size should be derived from the provided value, by providing the length value of the intended source, multiplied by the size of that length. Failure to perform the correct arithmetic to obtain the exact size of the value will likely result in the source overflowing its destination.

## General Recommendations

### How to avoid it

- Always perform the correct arithmetic to determine size.
- Specifically for memory allocation, calculate the allocation size from the allocation source:
    - Derive the size value from the length of intended source to determine the amount of units to be processed.
    - Always programmatically consider the size of the each unit and their conversion to memory units - for example, by using sizeof() on the unit's type.
    - Memory allocation should be a multiplication of the amount of units being written, times the size of each unit.

## Source Code Examples

### CPP

### Allocating and Assigning Memory without Sizeof Arithmetic

```
int *ptr;
ptr = (int*)malloc(5);
for (int i = 0; i < 5; i++)
{
      ptr[i] = i * 2 + 1;
}
```

### Allocating and Assigning Memory with Sizeof Arithmetic

```
int *ptr;
ptr = (int*)malloc(5 * sizeof(int));
for (int i = 0; i < 5; i++)
{
      ptr[i] = i * 2 + 1;
}
```

### Incorrect Arithmetic of Multi-Byte String Allocation

```
wchar_t * dest;
dest = (wchar_t *)malloc(wcslen(source) + 1); // Would not crash for a short "source"
wcscpy((wchar_t *)dest, source);
wprintf(L"Dest: %s\r\n", dest);
```

### Correct Arithmetic of Multi-Byte String Allocation

```
wchar_t * dest;
dest = (wchar_t *)malloc((wcslen(source) + 1) * sizeof(wchar_t));
wcscpy((wchar_t *)dest, source);
wprintf(L"Dest: %s\r\n", dest);
```

## Use of Function with Inconsistent Implementations

**Weakness ID:** 474 *(Weakness Base)* **Status:** Draft

### Description

### Description Summary

The code uses a function that has inconsistent implementations across operating systems and versions, which might cause security-relevant portability problems.

### Time of Introduction

- Architecture and Design
- Implementation

### Applicable Platforms

### Languages

C: *(Often)*

PHP: *(Often)*

All

### Potential Mitigations

Do not accept inconsistent behavior from the API specifications when the deviant behavior increase the risk level.

### Other Notes

The behavior of functions in this category varies by operating system, and at times, even by operating system version. Implementation differences can include:

- Slight differences in the way parameters are interpreted leading to inconsistent results.

- Some implementations of the function carry significant security risks.

- The function might not be defined on all platforms.

### Relationships

| Nature | Type | ID | Name | View(s) this relationship pertains to |
|---|---|---|---|---|
| ChildOf | Weakness Class | 398 | Indicator of Poor Code Quality | **Development Concepts (primary)699**<br>**Seven Pernicious Kingdoms (primary)700**<br>**Research Concepts (primary)1000** |
| ParentOf | Weakness Variant | 589 | Call to Non-ubiquitous API | **Research Concepts (primary)1000** |

### Taxonomy Mappings

| Mapped Taxonomy Name | Node ID | Fit | Mapped Node Name |
|---|---|---|---|
| 7 Pernicious Kingdoms | | | Inconsistent Implementations |

### 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 Potential Mitigations, Time of Introduction | | |
| 2008-09-08 | CWE Content Team | MITRE | Internal |
| | updated Applicable Platforms, Relationships, Other Notes, Taxonomy Mappings | | |

**Previous Entry Names**

| Change Date | Previous Entry Name |
|---|---|
| 2008-04-11 | Inconsistent Implementations |

BACK TO TOP

# Reliance on DNS Lookups in a Decision

## Risk

### What might happen

Relying on reverse DNS records, without verifying domain ownership via cryptographic certificates or protocols, is not a sufficient authentication mechanism. Basing any security decisions on the registered hostname could allow an external attacker to control the application flow. The attacker could possibly perform restricted operations, bypass access controls, and even spoof the user's identity, inject a bogus hostname into the security log, and possibly other logic attacks.

## Cause

### How does it happen

The application performs a reverse DNS resolution, based on the remote IP address, and performs a security check based on the returned hostname. However, it is relatively easy to spoof DNS names, or cause them to be misreported, depending on the context of the specific environment. If the remote server is controlled by the attacker, it can be configured to report a bogus hostname. Additionally, the attacker could also spoof the hostname if she controls the associated DNS server, or by attacking the legitimate DNS server, or by poisoning the server's DNS cache, or by modifying unprotected DNS traffic to the server. Regardless of the vector, a remote attacker can alter the detected network address, faking the authentication details.

## General Recommendations

### How to avoid it

- Do not rely on DNS records, network addresses, or system hostnames as a form of authentication, or any other security-related decision.
- Do not perform reverse DNS resolution over an unprotected protocol without record validation.
- Implement a proper authentication mechanism, such as passwords, cryptographic certificates, or public key digital signatures.
- Consider using proposed protocol extensions to cryptographically protect DNS, e.g. DNSSEC (though note the limited support and other drawbacks).

## Source Code Examples

### Java

### Using Reverse DNS as Authentication

```
private boolean isInternalEmployee(ServletRequest req) {
      boolean isCompany = false;

      String ip = req.getRemoteAddr();
      InetAddress address = InetAddress.getByName(ip);

      if (address.getHostName().endsWith(COMPANYNAME)) {
             isCompany = true;
     }
      return isCompany;
}
```

### Verify Authenticated User's Identity

```
private boolean isInternalEmployee(ServletRequest req) {
      boolean isCompany = false;

      Principal user = req.getUserPrincipal();
      if (user != null) {
      if (user.getName().startsWith(COMPANYDOMAIN + "\\")) {
             isCompany = true;
         }
    }
      return isCompany;
}
```

## Incorrect Permission Assignment for Critical Resource

**Weakness ID:** 732 *(Weakness Class)* **Status:** Draft

### Description

#### Description Summary

The software specifies permissions for a security-critical resource in a way that allows that resource to be read or modified by unintended actors.

#### Extended Description

When a resource is given a permissions setting that provides access to a wider range of actors than required, it could lead to the disclosure of sensitive information, or the modification of that resource by unintended parties. This is especially dangerous when the resource is related to program configuration, execution or sensitive user data.

### Time of Introduction

- Architecture and Design
- Implementation
- Installation
- Operation

### Applicable Platforms

#### Languages

Language-independent

### Modes of Introduction

The developer may set loose permissions in order to minimize problems when the user first runs the program, then create documentation stating that permissions should be tightened. Since system administrators and users do not always read the documentation, this can result in insecure permissions being left unchanged.

The developer might make certain assumptions about the environment in which the software runs - e.g., that the software is running on a single-user system, or the software is only accessible to trusted administrators. When the software is running in a different environment, the permissions become a problem.

### Common Consequences

| Scope | Effect |
|---|---|
| Confidentiality | An attacker may be able to read sensitive information from the associated resource, such as credentials or configuration information stored in a file. |
| Integrity | An attacker may be able to modify critical properties of the associated resource to gain privileges, such as replacing a world-writable executable with a Trojan horse. |
| Availability | An attacker may be able to destroy or corrupt critical data in the associated resource, such as deletion of records from a database. |

### Likelihood of Exploit

Medium to High

### Detection Methods

#### Automated Static Analysis

Automated static analysis may be effective in detecting permission problems for system resources such as files, directories, shared memory, device interfaces, etc. Automated techniques may be able to detect the use of library functions that modify permissions, then analyze function calls for arguments that contain potentially insecure values.

However, since the software's intended security policy might allow loose permissions for certain operations (such as publishing a file on a web server), automated static analysis may produce some false positives - i.e., warnings that do not have any security consequences or require any code changes.

When custom permissions models are used - such as defining who can read messages in a particular forum in a bulletin board system - these can be difficult to detect using automated static analysis. It may be possible to define custom signatures that identify any custom functions that implement the permission checks and assignments.

### Automated Dynamic Analysis

Automated dynamic analysis may be effective in detecting permission problems for system resources such as files, directories, shared memory, device interfaces, etc.

However, since the software's intended security policy might allow loose permissions for certain operations (such as publishing a file on a web server), automated dynamic analysis may produce some false positives - i.e., warnings that do not have any security consequences or require any code changes.

When custom permissions models are used - such as defining who can read messages in a particular forum in a bulletin board system - these can be difficult to detect using automated dynamic analysis. It may be possible to define custom signatures that identify any custom functions that implement the permission checks and assignments.

### Manual Static Analysis

Manual static analysis may be effective in detecting the use of custom permissions models and functions. The code could then be examined to identifying usage of the related functions. Then the human analyst could evaluate permission assignments in the context of the intended security model of the software.

### Manual Dynamic Analysis

Manual dynamic analysis may be effective in detecting the use of custom permissions models and functions. The program could then be executed with a focus on exercising code paths that are related to the custom permissions. Then the human analyst could evaluate permission assignments in the context of the intended security model of the software.

### Fuzzing

Fuzzing is not effective in detecting this weakness.

## Demonstrative Examples

## Example 1

The following code sets the umask of the process to 0 before creating a file and writing "Hello world" into the file.

*(Bad Code)*

*Example Language:* **C**

```
#define OUTFILE "hello.out"

umask(0);
FILE *out;
/* Ignore CWE-59 (link following) for brevity */
out = fopen(OUTFILE, "w");
if (out) {
fprintf(out, "hello world!\n");
fclose(out);
}
```

After running this program on a UNIX system, running the "ls -l" command might return the following output:

*(Result)*

```
-rw-rw-rw- 1 username 13 Nov 24 17:58 hello.out
```

The "rw-rw-rw-" string indicates that the owner, group, and world (all users) can read the file and write to it.

## Example 2

The following code snippet might be used as a monitor to periodically record whether a web site is alive. To ensure that the file can always be modified, the code uses chmod() to make the file world-writable.

*(Bad Code)*

*Example Language:* **Perl**

```
$fileName = "secretFile.out";

if (-e $fileName) {
chmod 0777, $fileName;
}
```

```
my $outFH;
if (! open($outFH, ">>$fileName")) {
ExitError("Couldn't append to $fileName: $!");
}
my $dateString = FormatCurrentTime();
my $status = IsHostAlive("cwe.mitre.org");
print $outFH "$dateString cwe status: $status!\n";
close($outFH);
```

The first time the program runs, it might create a new file that inherits the permissions from its environment. A file listing might look like:

*(Result)*

```
-rw-r--r-- 1 username 13 Nov 24 17:58 secretFile.out
```

This listing might occur when the user has a default umask of 022, which is a common setting. Depending on the nature of the file, the user might not have intended to make it readable by everyone on the system.

The next time the program runs, however - and all subsequent executions - the chmod will set the file's permissions so that the owner, group, and world (all users) can read the file and write to it:

*(Result)*

```
-rw-rw-rw- 1 username 13 Nov 24 17:58 secretFile.out
```

Perhaps the programmer tried to do this because a different process uses different permissions that might prevent the file from being updated.

## Example 3

The following command recursively sets world-readable permissions for a directory and all of its children:

*(Bad Code)*

*Example Language:* **Shell**

```
chmod -R ugo+r DIRNAME
```

If this command is run from a program, the person calling the program might not expect that all the files under the directory will be world-readable. If the directory is expected to contain private data, this could become a security problem.

### Observed Examples

| Reference | Description |
|---|---|
| CVE-2009-3482 | Anti-virus product sets insecure "Everyone: Full Control" permissions for files under the "Program Files" folder, allowing attackers to replace executables with Trojan horses. |
| CVE-2009-3897 | Product creates directories with 0777 permissions at installation, allowing users to gain privileges and access a socket used for authentication. |
| CVE-2009-3489 | Photo editor installs a service with an insecure security descriptor, allowing users to stop or start the service, or execute commands as SYSTEM. |
| CVE-2009-3289 | Library function copies a file to a new target and uses the source file's permissions for the target, which is incorrect when the source file is a symbolic link, which typically has 0777 permissions. |
| CVE-2009-0115 | Device driver uses world-writable permissions for a socket file, allowing attackers to inject arbitrary commands. |
| CVE-2009-1073 | LDAP server stores a cleartext password in a world-readable file. |
| CVE-2009-0141 | Terminal emulator creates TTY devices with world-writable permissions, allowing an attacker to write to the terminals of other users. |

| | |
|---|---|
| CVE-2008-0662 | VPN product stores user credentials in a registry key with "Everyone: Full Control" permissions, allowing attackers to steal the credentials. |
| CVE-2008-0322 | Driver installs its device interface with "Everyone: Write" permissions. |
| CVE-2009-3939 | Driver installs a file with world-writable permissions. |
| CVE-2009-3611 | Product changes permissions to 0777 before deleting a backup; the permissions stay insecure for subsequent backups. |
| CVE-2007-6033 | Product creates a share with "Everyone: Full Control" permissions, allowing arbitrary program execution. |
| CVE-2007-5544 | Product uses "Everyone: Full Control" permissions for memory-mapped files (shared memory) in inter-process communication, allowing attackers to tamper with a session. |
| CVE-2005-4868 | Database product uses read/write permissions for everyone for its shared memory, allowing theft of credentials. |
| CVE-2004-1714 | Security product uses "Everyone: Full Control" permissions for its configuration files. |
| CVE-2001-0006 | "Everyone: Full Control" permissions assigned to a mutex allows users to disable network connectivity. |
| CVE-2002-0969 | Chain: database product contains buffer overflow that is only reachable through a .ini configuration file - which has "Everyone: Full Control" permissions. |

## Potential Mitigations

### Phase: Implementation

When using a critical resource such as a configuration file, check to see if the resource has insecure permissions (such as being modifiable by any regular user), and generate an error or even exit the software if there is a possibility that the resource could have been modified by an unauthorized party.

### Phase: Architecture and Design

Divide your application into anonymous, normal, privileged, and administrative areas. Reduce the attack surface by carefully defining distinct user groups, privileges, and/or roles. Map these against data, functionality, and the related resources. Then set the permissions accordingly. This will allow you to maintain more fine-grained control over your resources.

### Phases: Implementation; Installation

During program startup, explicitly set the default permissions or umask to the most restrictive setting possible. Also set the appropriate permissions during program installation. This will prevent you from inheriting insecure permissions from any user who installs or runs the program.

### Phase: System Configuration

For all configuration files, executables, and libraries, make sure that they are only readable and writable by the software's administrator.

### Phase: Documentation

Do not suggest insecure configuration changes in your documentation, especially if those configurations can extend to resources and other software that are outside the scope of your own software.

### Phase: Installation

Do not assume that the system administrator will manually change the configuration to the settings that you recommend in the manual.

### Phase: Testing

Use tools and techniques that require manual (human) analysis, such as penetration testing, threat modeling, and interactive tools that allow the tester to record and modify an active session. These may be more effective than strictly automated techniques. This is especially the case with weaknesses that are related to design and business rules.

### Phase: Testing

Use monitoring tools that examine the software's process as it interacts with the operating system and the network. This technique is useful in cases when source code is unavailable, if the software was not developed by you, or if you want to verify that the build phase did not introduce any new weaknesses. Examples include debuggers that directly attach to the running process; system-call tracing utilities such as truss (Solaris) and strace (Linux); system activity monitors such as FileMon, RegMon, Process Monitor, and other Sysinternals utilities (Windows); and sniffers and protocol analyzers that monitor network traffic.

Attach the monitor to the process and watch for library functions or system calls on OS resources such as files, directories, and shared memory. Examine the arguments to these calls to infer which permissions are being used.

Note that this technique is only useful for permissions issues related to system resources. It is not likely to detect application-level business rules that are related to permissions, such as if a user of a blog system marks a post as "private," but the blog system inadvertently marks it as "public."

**Phases: Testing; System Configuration**

Ensure that your software runs properly under the Federal Desktop Core Configuration (FDCC) or an equivalent hardening configuration guide, which many organizations use to limit the attack surface and potential risk of deployed software.

## Relationships

| Nature | Type | ID | Name | View(s) this relationship pertains to |
|---|---|---|---|---|
| ChildOf | Category | 275 | Permission Issues | **Development Concepts (primary)699** |
| ChildOf | Weakness Class | 668 | Exposure of Resource to Wrong Sphere | **Research Concepts (primary)1000** |
| ChildOf | Category | 753 | 2009 Top 25 - Porous Defenses | **Weaknesses in the 2009 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)750** |
| ChildOf | Category | 803 | 2010 Top 25 - Porous Defenses | **Weaknesses in the 2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800** |
| RequiredBy | Compound Element: Composite | 689 | Permission Race Condition During Resource Copy | Research Concepts1000 |
| ParentOf | Weakness Variant | 276 | Incorrect Default Permissions | **Research Concepts (primary)1000** |
| ParentOf | Weakness Variant | 277 | Insecure Inherited Permissions | **Research Concepts (primary)1000** |
| ParentOf | Weakness Variant | 278 | Insecure Preserved Inherited Permissions | **Research Concepts (primary)1000** |
| ParentOf | Weakness Variant | 279 | Incorrect Execution-Assigned Permissions | **Research Concepts (primary)1000** |
| ParentOf | Weakness Base | 281 | Improper Preservation of Permissions | **Research Concepts (primary)1000** |

## Related Attack Patterns

| CAPEC-ID | Attack Pattern Name | *(CAPEC Version: 1.5)* |
|---|---|---|
| 232 | Exploitation of Privilege/Trust | |
| 1 | Accessing Functionality Not Properly Constrained by ACLs | |
| 17 | Accessing, Modifying or Executing Executable Files | |
| 60 | Reusing Session IDs (aka Session Replay) | |
| 61 | Session Fixation | |
| 62 | Cross Site Request Forgery (aka Session Riding) | |
| 122 | Exploitation of Authorization | |
| 180 | Exploiting Incorrectly Configured Access Control Security Levels | |
| 234 | Hijacking a privileged process | |

## References

Mark Dowd, John McDonald and Justin Schuh. "The Art of Software Security Assessment". Chapter 9, "File Permissions." Page 495.. 1st Edition. Addison Wesley. 2006.

John Viega and Gary McGraw. "Building Secure Software". Chapter 8, "Access Control." Page 194.. 1st Edition. Addison-Wesley. 2002.

### Maintenance Notes

The relationships between privileges, permissions, and actors (e.g. users and groups) need further refinement within the Research view. One complication is that these concepts apply to two different pillars, related to control of resources (CWE-664) and protection mechanism failures (CWE-396).

## Content History

**Submissions**

| Submission Date | Submitter | Organization | Source |
|---|---|---|---|
| 2008-09-08 | | | Internal CWE Team |
| | new weakness-focused entry for Research view. | | |

**Modifications**

| Modification Date | Modifier | Organization | Source |
|---|---|---|---|
| 2009-01-12 | CWE Content Team | MITRE | Internal |
| | updated Description, Likelihood of Exploit, Name, Potential Mitigations, Relationships | | |
| 2009-03-10 | CWE Content Team | MITRE | Internal |
| | updated Potential Mitigations, Related Attack Patterns | | |
| 2009-05-27 | CWE Content Team | MITRE | Internal |
| | updated Name | | |
| 2009-12-28 | CWE Content Team | MITRE | Internal |
| | updated Applicable Platforms, Common Consequences, Demonstrative Examples, Detection Factors, Modes of Introduction, Observed Examples, Potential Mitigations, References | | |
| 2010-02-16 | CWE Content Team | MITRE | Internal |
| | updated Relationships | | |
| 2010-04-05 | CWE Content Team | MITRE | Internal |
| | updated Potential Mitigations, Related Attack Patterns | | |

**Previous Entry Names**

| Change Date | Previous Entry Name |
|---|---|
| 2009-01-12 | Insecure Permission Assignment for Resource |
| 2009-05-27 | Insecure Permission Assignment for Critical Resource |

BACK TO TOP

# Use of Insufficiently Random Values

## Risk

**What might happen**

Random values are often used as a mechanism to prevent malicious users from guessing a value, such as a password, encryption key, or session identifier. Depending on what this random value is used for, an attacker would be able to predict the next numbers generated, or previously generated values. This could enable the attacker to hijack another user's session, impersonate another user, or crack an encryption key (depending on what the pseudo-random value was used for).

## Cause

**How does it happen**

The application uses a weak method of generating pseudo-random values, such that other numbers could be determined from a relatively small sample size. Since the pseudo-random number generator used is designed for statistically uniform distribution of values, it is approximately deterministic. Thus, after collecting a few generated values (e.g. by creating a few individual sessions, and collecting the sessionids), it would be possible for an attacker to calculate another sessionid.

Specifically, if this pseudo-random value is used in any security context, such as passwords, keys, or secret identifiers, an attacker would be able to predict the next numbers generated, or previously generated values.

## General Recommendations

**How to avoid it**

Generic Guidance:

- Whenever unpredicatable numbers are required in a security context, use a cryptographically strong random number generator, instead of a statistical pseudo-random generator.
- Use the cryptorandom generator that is built-in to your language or platform, and ensure it is securely seeded. Do not seed the generator with a weak, non-random seed. (In most cases, the default is securely random).
- Ensure you use a long enough random value, to make brute-force attacks unfeasible.

Specific Recommendations:

- Do not use the statistical pseudo-random number generator, use the cryptorandom generator instead. In Java, this is the SecureRandom class.

## Source Code Examples

**Java**

**Use of a weak pseudo-random number generator**

```
Random random = new Random();

long sessNum = random.nextLong();

String sessionId = sessNum.toString();
```

### Cryptographically secure random number generator

```
SecureRandom random = new SecureRandom();

byte sessBytes[] = new byte[32];

random.nextBytes(sessBytes);

String sessionId = new String(sessBytes);
```

## Objc

### Use of a weak pseudo-random number generator

```
long sessNum = rand();
NSString* sessionId = [NSString stringWithFormat:@"%ld", sessNum];
```

### Cryptographically secure random number generator

```
UInt32 sessBytes;
SecRandomCopyBytes(kSecRandomDefault, sizeof(sessBytes), (uint8_t*)&sessBytes);

NSString* sessionId = [NSString stringWithFormat:@"%llu", sessBytes];
```

## Swift

### Use of a weak pseudo-random number generator

```
let sessNum = rand();
let sessionId = String(format:"%ld", sessNum)
```

### Cryptographically secure random number generator

```
var sessBytes: UInt32 = 0
withUnsafeMutablePointer(&sessBytes, { (sessBytesPointer) -> Void in
    let castedPointer = unsafeBitCast(sessBytesPointer, UnsafeMutablePointer<UInt8>.self)
    SecRandomCopyBytes(kSecRandomDefault, sizeof(UInt32), castedPointer)
})

let sessionId = String(format:"%llu", sessBytes)
```

# Unchecked Return Value

## Risk

### What might happen

A program that does not check function return values could cause the application to enter an undefined state. This could lead to unexpected behavior and unintended consequences, including inconsistent data, system crashes or other error-based exploits.

## Cause

### How does it happen

The application calls a system function, but does not receive or check the result of this funciton. These functions often return error codes in the result, or share other status codes with it's caller. The application simply ignores this result value, losing this vital information.

## General Recommendations

### How to avoid it

- Always check the result of any called function that returns a value, and verify the result is an expected value.

- Ensure the calling function responds to all possible return values.

- Expect runtime errors and handle them gracefully. Explicitly define a mechanism for handling unexpected errors.

## Source Code Examples

### CPP

### Unchecked Memory Allocation

```
buff = (char*) malloc(size);
strncpy(buff, source, size);
```

### Safer Memory Allocation

```
buff = (char*) malloc(size+1);
if (buff==NULL) exit(1);

strncpy(buff, source, size);
buff[size] = '\0';
```

## Use of sizeof() on a Pointer Type

**Weakness ID:** 467 *(Weakness Variant)* **Status:** Draft

### Description

## Description Summary

The code calls sizeof() on a malloced pointer type, which always returns the wordsize/8. This can produce an unexpected result if the programmer intended to determine how much memory has been allocated.

### Time of Introduction

- Implementation

### Applicable Platforms

## Languages

C

C++

### Common Consequences

| Scope | Effect |
|---|---|
| Integrity | This error can often cause one to allocate a buffer that is much smaller than what is needed, leading to resultant weaknesses such as buffer overflows. |

### Likelihood of Exploit

High

### Demonstrative Examples

## Example 1

Care should be taken to ensure sizeof returns the size of the data structure itself, and not the size of the pointer to the data structure.

In this example, sizeof(foo) returns the size of the pointer.

*(Bad Code)*

*Example Languages:* **C and C++**

```
double *foo;
...
foo = (double *)malloc(sizeof(foo));
```

In this example, sizeof(*foo) returns the size of the data structure and not the size of the pointer.

*(Good Code)*

*Example Languages:* **C and C++**

```
double *foo;
...
foo = (double *)malloc(sizeof(*foo));
```

## Example 2

This example defines a fixed username and password. The AuthenticateUser() function is intended to accept a username and a password from an untrusted user, and check to ensure that it matches the username and password. If the username and password match, AuthenticateUser() is intended to indicate that authentication succeeded.

*(Bad Code)*

```
/* Ignore CWE-259 (hard-coded password) and CWE-309 (use of password system for authentication) for this example. */

char *username = "admin";
char *pass = "password";

int AuthenticateUser(char *inUser, char *inPass) {
```

```
printf("Sizeof username = %d\n", sizeof(username));
printf("Sizeof pass = %d\n", sizeof(pass));

if (strncmp(username, inUser, sizeof(username))) {
printf("Auth failure of username using sizeof\n");
return(AUTH_FAIL);
}
/* Because of CWE-467, the sizeof returns 4 on many platforms and architectures. */
if (! strncmp(pass, inPass, sizeof(pass))) {
printf("Auth success of password using sizeof\n");
return(AUTH_SUCCESS);
}
else {
printf("Auth fail of password using sizeof\n");
return(AUTH_FAIL);
}
}

int main (int argc, char **argv)
{
int authResult;

if (argc < 3) {
ExitError("Usage: Provide a username and password");
}
authResult = AuthenticateUser(argv[1], argv[2]);
if (authResult != AUTH_SUCCESS) {
ExitError("Authentication failed");
}
else {
DoAuthenticatedTask(argv[1]);
}
}
```

In AuthenticateUser(), because sizeof() is applied to a parameter with an array type, the sizeof() call might return 4 on many modern architectures. As a result, the strncmp() call only checks the first four characters of the input password, resulting in a partial comparison (CWE-187), leading to improper authentication (CWE-287).

Because of the partial comparison, any of these passwords would still cause authentication to succeed for the "admin" user:

*(Attack)*

```
pass5
passABCDEFGH
passWORD
```

Because only 4 characters are checked, this significantly reduces the search space for an attacker, making brute force attacks more feasible.

The same problem also applies to the username, so values such as "adminXYZ" and "administrator" will succeed for the username.

## Potential Mitigations

### Phase: Implementation

Use expressions such as "sizeof(*pointer)" instead of "sizeof(pointer)", unless you intend to run sizeof() on a pointer type to gain some platform independence or if you are allocating a variable on the stack.

## Other Notes

The use of sizeof() on a pointer can sometimes generate useful information. An obvious case is to find out the wordsize on a platform. More often than not, the appearance of sizeof(pointer) indicates a bug.

## Weakness Ordinalities

| Ordinality | Description |
|---|---|
| Primary | *(where the weakness exists independent of other weaknesses)* |

## Relationships

| Nature | Type | ID | Name | View(s) this relationship pertains to |
|---|---|---|---|---|
| ChildOf | Category | 465 | Pointer Issues | **Development Concepts (primary)699** |
| ChildOf | Weakness Class | 682 | Incorrect Calculation | **Research Concepts (primary)1000** |
| ChildOf | Category | 737 | CERT C Secure Coding Section 03 - Expressions (EXP) | **Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734** |
| ChildOf | Category | 740 | CERT C Secure Coding Section 06 - Arrays (ARR) | Weaknesses Addressed by the CERT C Secure Coding Standard734 |
| CanPrecede | Weakness Base | 131 | Incorrect Calculation of Buffer Size | Research Concepts1000 |

## Taxonomy Mappings

| Mapped Taxonomy Name | Node ID | Fit | Mapped Node Name |
|---|---|---|---|
| CLASP | | | Use of sizeof() on a pointer type |
| CERT C Secure Coding | ARR01-C | | Do not apply the sizeof operator to a pointer when taking the size of an array |
| CERT C Secure Coding | EXP01-C | | Do not take the size of a pointer to determine the size of the pointed-to type |

## White Box Definitions

A weakness where code path has:

1. end statement that passes an identity of a dynamically allocated memory resource to a sizeof operator
2. start statement that allocates the dynamically allocated memory resource

## References

Robert Seacord. "EXP01-A. Do not take the sizeof a pointer to determine the size of a type". <https://www.securecoding.cert.org/confluence/display/seccode/EXP01-A.+Do+not+take+the+sizeof+a+pointer+to+determine+the+size+of+a+type>.

## Content History

### Submissions

| Submission Date | Submitter | Organization | Source |
|---|---|---|---|
| | CLASP | | Externally Mined |

### Modifications

| Modification Date | Modifier | Organization | Source |
|---|---|---|---|
| 2008-07-01 | Eric Dalci | Cigital | External |
| | updated Time of Introduction | | |
| 2008-08-01 | | KDM Analytics | External |
| | added/updated white box definitions | | |
| 2008-09-08 | CWE Content Team | MITRE | Internal |
| | updated Applicable Platforms, Common Consequences, Relationships, Other Notes, Taxonomy Mappings, Weakness Ordinalities | | |
| 2008-11-24 | CWE Content Team | MITRE | Internal |
| | updated Relationships, Taxonomy Mappings | | |
| 2009-03-10 | CWE Content Team | MITRE | Internal |
| | updated Demonstrative Examples | | |
| 2009-12-28 | CWE Content Team | MITRE | Internal |
| | updated Demonstrative Examples | | |
| 2010-02-16 | CWE Content Team | MITRE | Internal |
| | updated Relationships | | |

BACK TO TOP

# NULL Pointer Dereference

## Risk

**What might happen**

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

## Cause

**How does it happen**

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

## General Recommendations

**How to avoid it**

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

## Source Code Examples

## Improper Validation of Array Index

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

## Description

### Description Summary

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

## Alternate Terms

**out-of-bounds array index**

**index-out-of-range**

**array index underflow**

## Time of Introduction

- Implementation

## Applicable Platforms

### Languages

C: *(Often)*

C++: *(Often)*

Language-independent

## Common Consequences

| Scope | Effect |
|---|---|
| Integrity<br>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<br>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<br>Availability<br>Confidentiality | A single fault could allow either an overflow (CWE-788) or underflow (CWE-786) of the array index. What happens next will depend on the type of operation being performed out of bounds, but can expose sensitive information, cause a system crash, or possibly lead to arbitrary code execution. |

## Likelihood of Exploit

High

## Detection Methods

### Automated Static Analysis

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

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

### *Effectiveness: High*

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

### Automated Dynamic Analysis

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

### Black Box

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

## Demonstrative Examples

## Example 1

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

*(Bad Code)*

*Example Language:* **C**

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

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

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

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

*(Good Code)*

*Example Language:* **C**

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

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

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

## Example 2

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

*(Bad Code)*

*Example Language:* **Java**

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

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

## Example 3

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

*(Bad Code)*

*Example Language:* **Java**

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

String productSummary = new String("");

try {
String productSummary = getProductSummary(index);

} catch (Exception ex) {...}

return productSummary;
}

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

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

*(Good Code)*

*Example Language:* **Java**

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

String productSummary = new String("");
try {
String productSummary = getProductSummary(index);

} catch (Exception ex) {...}

return productSummary;
}

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

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

return productSummary;
}
```

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

*(Good Code)*

*Example Language:* **Java**

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

## Observed Examples

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

## Potential Mitigations

**Phase: Architecture and Design**

### Strategies: Input Validation; Libraries or Frameworks

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

**Phase: Architecture and Design**

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

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

**Phase: Requirements**

### Strategy: Language Selection

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

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

**Phase: Implementation**

### Strategy: Input Validation

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

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

**Phase: Implementation**

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

## Weakness Ordinalities

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

## Relationships

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

## Theoretical Notes

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

## Affected Resources

- Memory

## f Causal Nature

Explicit

## Taxonomy Mappings

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

## Related Attack Patterns

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

## References

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

## Content History

**Submissions**

| Submission Date | Submitter | Organization | Source |
|---|---|---|---|
| | CLASP | | Externally Mined |

**Modifications**

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

**Previous Entry Names**

| Change Date | Previous Entry Name |
|---|---|
| 2009-10-29 | Unchecked Array Indexing |

BACK TO TOP

## Scanned Languages

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