# vul_files_12 Scan Report

| | |
|---|---|
| Project Name | vul_files_12 |
| Scan Start | Monday, January 6, 2025 7:49:35 PM |
| Preset | Checkmarx Default |
| Scan Time | 02h:35m:57s |
| Lines Of Code Scanned | 299493 |
| Files Scanned | 226 |
| Report Creation Time | Monday, January 6, 2025 10:50:49 PM |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14 |
| Team | CxServer |
| Checkmarx Version | 8.7.0 |
| Scan Type | Full |
| Source Origin | LocalPath |
| Density | 5/1000 (Vulnerabilities/LOC) |
| Visibility | Public |

## Filter Settings

**Severity**

Included: High, Medium, Low, Information

Excluded: None

**Result State**

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

Excluded: None

**Assigned to**

Included: All

**Categories**

Included:

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

Excluded:

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

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

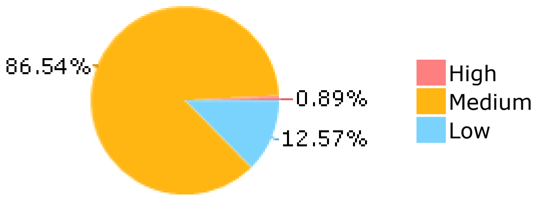
**Results Limit**

Results limit per query was set to 50

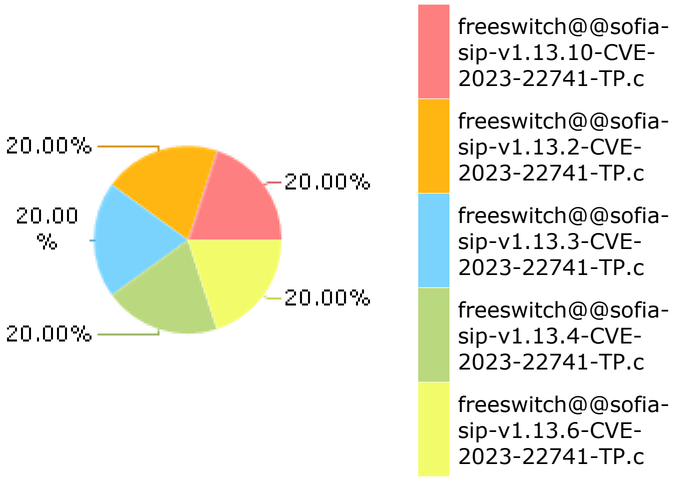
**Selected Queries**

Selected queries are listed in Result Summary

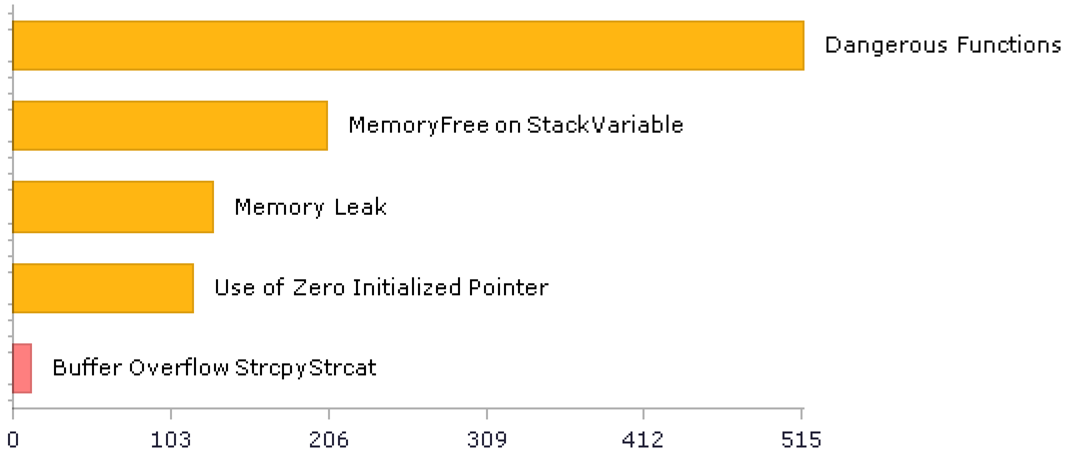
## Result Summary



## Most Vulnerable Files



## Top 5 Vulnerabilities



# Scan Summary - OWASP Top 10 2017

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

| Category | Threat Agent | Exploitability | Weakness Prevalence | Weakness Detectability | Technical Impact | Business Impact | Issues Found | Best Fix Locations |
|---|---|---|---|---|---|---|---|---|
| A1-Injection | App. Specific | EASY | COMMON | EASY | SEVERE | App. Specific | 127 | 112 |
| A2-Broken Authentication | App. Specific | EASY | COMMON | AVERAGE | SEVERE | App. Specific | 6 | 6 |
| A3-Sensitive Data Exposure | App. Specific | AVERAGE | WIDESPREAD | AVERAGE | SEVERE | App. Specific | 16 | 16 |
| 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 | 516 | 516 |
| 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 | 16 | 16 |
| 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 | 516 | 516 |
| 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 | 109 | 109 |
| 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. | 6 | 6 |
| 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. | 5 | 5 |
| 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. | 36 | 31 |
| 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. | 3 | 3 |

* 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) | 6 | 6 |
| AC-4 Information Flow Enforcement (P1) | 0 | 0 |
| AC-6 Least Privilege (P1) | 0 | 0 |
| AU-9 Protection of Audit Information (P1) | 0 | 0 |
| CM-6 Configuration Settings (P2) | 0 | 0 |
| IA-5 Authenticator Management (P1) | 0 | 0 |
| IA-6 Authenticator Feedback (P2) | 0 | 0 |
| IA-8 Identification and Authentication (Non-Organizational Users) (P1) | 0 | 0 |
| SC-12 Cryptographic Key Establishment and Management (P1) | 0 | 0 |
| SC-13 Cryptographic Protection (P1) | 20 | 15 |
| SC-17 Public Key Infrastructure Certificates (P1) | 0 | 0 |
| SC-18 Mobile Code (P2) | 0 | 0 |
| SC-23 Session Authenticity (P1)* | 0 | 0 |
| SC-28 Protection of Information at Rest (P1) | 5 | 5 |
| SC-4 Information in Shared Resources (P1) | 16 | 16 |
| SC-5 Denial of Service Protection (P1)* | 280 | 215 |
| SC-8 Transmission Confidentiality and Integrity (P1) | 0 | 0 |
| SI-10 Information Input Validation (P1)* | 65 | 65 |
| SI-11 Error Handling (P2)* | 77 | 77 |
| SI-15 Information Output Filtering (P0) | 0 | 0 |
| SI-16 Memory Protection (P1) | 0 | 0 |

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

# Scan Summary - OWASP Mobile Top 10 2016

| Category | Description | Issues Found | Best Fix Locations |
|---|---|---|---|
| M1-Improper Platform Usage | This category covers misuse of a platform feature or failure to use platform security controls. It might include Android intents, platform permissions, misuse of TouchID, the Keychain, or some other security control that is part of the mobile operating system. There are several ways that mobile apps can experience this risk. | 0 | 0 |
| M2-Insecure Data Storage | This category covers insecure data storage and unintended data leakage. | 0 | 0 |
| M3-Insecure Communication | This category covers poor handshaking, incorrect SSL versions, weak negotiation, cleartext communication of sensitive assets, etc. | 0 | 0 |
| M4-Insecure Authentication | This category captures notions of authenticating the end user or bad session management. This can include:<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

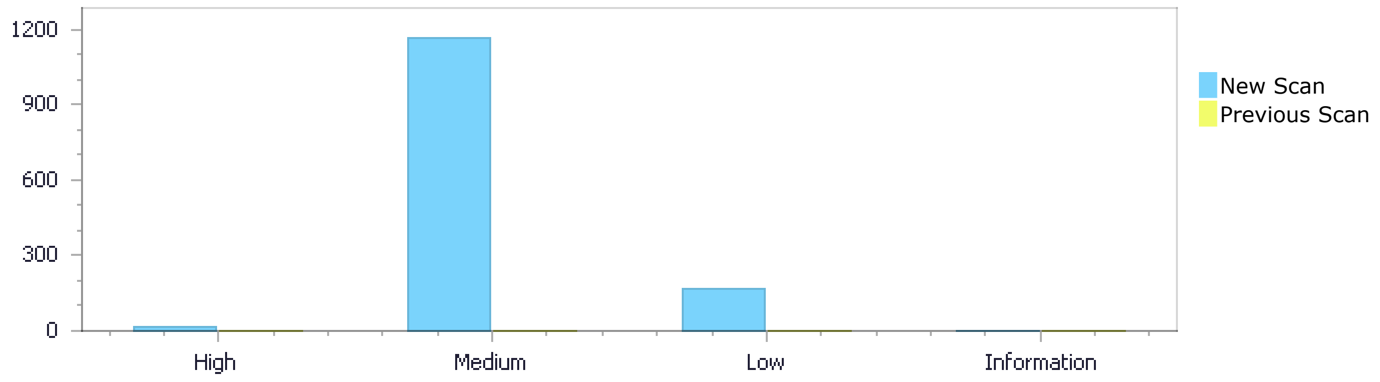
| Category | Issues Found | Best Fix Locations |
|---|---|---|
| Must audit | 0 | 0 |
| Check | 0 | 0 |
| Optional | 0 | 0 |

## Results Distribution By Status

First scan of the project

| | High | Medium | Low | Information | Total |
|---|---|---|---|---|---|
| New Issues | 12 | 1,170 | 170 | 0 | 1,352 |
| Recurrent Issues | 0 | 0 | 0 | 0 | 0 |
| Total | 12 | 1,170 | 170 | 0 | 1,352 |

| | | | | | |
|---|---|---|---|---|---|
| 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 | 12 | 1,170 | 170 | 0 | 1,352 |
| Urgent | 0 | 0 | 0 | 0 | 0 |
| Proposed Not Exploitable | 0 | 0 | 0 | 0 | 0 |
| Total | 12 | 1,170 | 170 | 0 | 1,352 |

## Result Summary

| Vulnerability Type | Occurrences | Severity |
|---|---|---|
| Buffer Overflow StrcpyStrcat | 12 | High |
| Dangerous Functions | 516 | Medium |
| MemoryFree on StackVariable | 205 | Medium |
| Memory Leak | 130 | Medium |
| Use of Zero Initialized Pointer | 117 | Medium |

| | | |
|---|---|---|
| Buffer Overflow boundcpy WrongSizeParam | 82 | Medium |
| Wrong Size t Allocation | 66 | Medium |
| Use of a One Way Hash without a Salt | 20 | Medium |
| Heap Inspection | 16 | Medium |
| Char Overflow | 12 | Medium |
| Divide By Zero | 3 | Medium |
| Integer Overflow | 3 | Medium |
| Unchecked Return Value | 77 | Low |
| Unchecked Array Index | 38 | Low |
| NULL Pointer Dereference | 33 | Low |
| Use of Sizeof On a Pointer Type | 11 | Low |
| Incorrect Permission Assignment For Critical Resources | 6 | Low |
| Information Exposure Through Comments | 5 | Low |

## 10 Most Vulnerable Files

### High and Medium Vulnerabilities

| File Name | Issues Found |
|---|---|
| freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | 53 |
| freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | 53 |
| freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | 53 |
| freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c | 53 |
| freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c | 53 |
| FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | 40 |
| FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | 40 |
| FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | 40 |
| FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | 32 |
| FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp | 32 |

# Scan Results Details

## 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=1000019&projectid=14&pathid=1 |
| Status | New |

The size of the buffer used by ntlm_current_time in timestamp, at line 186 of FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ntlm_current_time passes to timestamp, at line 186 of FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 186 | 193 |
| Object | timestamp | timestamp |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c
Method void ntlm_current_time(BYTE* timestamp)

```
....
186.  void ntlm_current_time(BYTE* timestamp)
....
193.        CopyMemory(timestamp, &(time64.QuadPart), 8);
```

**Buffer Overflow StrcpyStrcat\Path 2:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=2 |
| Status | New |

The size of the buffer used by ntlm_generate_signing_key in exported_session_key, at line 600 of FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ntlm_generate_signing_key passes to exported_session_key, at line 600 of FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 600 | 612 |
| Object | exported_session_key | exported_session_key |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c
Method static int ntlm_generate_signing_key(BYTE* exported_session_key, PSecBuffer sign_magic,

```
....
600.  static int ntlm_generate_signing_key(BYTE* exported_session_key,
PSecBuffer sign_magic,
....
612.         CopyMemory(value, exported_session_key,
WINPR_MD5_DIGEST_LENGTH);
```

### Buffer Overflow StrcpyStrcat\Path 3:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=3 |
| Status | New |

The size of the buffer used by ntlm_generate_sealing_key in exported_session_key, at line 661 of FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ntlm_generate_sealing_key passes to exported_session_key, at line 661 of FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 661 | 672 |
| Object | exported_session_key | exported_session_key |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c
Method static int ntlm_generate_sealing_key(BYTE* exported_session_key, PSecBuffer seal_magic,

```
....
661.  static int ntlm_generate_sealing_key(BYTE* exported_session_key,
PSecBuffer seal_magic,
....
672.         CopyMemory(p, exported_session_key,
WINPR_MD5_DIGEST_LENGTH);
```

**Buffer Overflow StrcpyStrcat\Path 4:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=4](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=4) |
| Status | New |

The size of the buffer used by ntlm_current_time in timestamp, at line 186 of FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ntlm_current_time passes to timestamp, at line 186 of FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c |
| Line | 186 | 193 |
| Object | timestamp | timestamp |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c |
| Method | void ntlm_current_time(BYTE* timestamp) |

```
....
186.  void ntlm_current_time(BYTE* timestamp)
....
193.        CopyMemory(timestamp, &(time64.QuadPart), 8);
```

**Buffer Overflow StrcpyStrcat\Path 5:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=5](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=5) |
| Status | New |

The size of the buffer used by ntlm_generate_signing_key in exported_session_key, at line 600 of FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ntlm_generate_signing_key passes to exported_session_key, at line 600 of FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c |
| Line | 600 | 612 |
| Object | exported_session_key | exported_session_key |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c |

| | |
|---|---|
| Method | static int ntlm_generate_signing_key(BYTE* exported_session_key, PSecBuffer sign_magic, |

```
....
600.  static int ntlm_generate_signing_key(BYTE* exported_session_key,
PSecBuffer sign_magic,
....
612.        CopyMemory(value, exported_session_key,
WINPR_MD5_DIGEST_LENGTH);
```

**Buffer Overflow StrcpyStrcat\Path 6:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=6 |
| Status | New |

The size of the buffer used by ntlm_generate_sealing_key in exported_session_key, at line 661 of FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ntlm_generate_sealing_key passes to exported_session_key, at line 661 of FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c |
| Line | 661 | 672 |
| Object | exported_session_key | exported_session_key |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c |
| Method | static int ntlm_generate_sealing_key(BYTE* exported_session_key, PSecBuffer seal_magic, |

```
....
661.  static int ntlm_generate_sealing_key(BYTE* exported_session_key,
PSecBuffer seal_magic,
....
672.        CopyMemory(p, exported_session_key,
WINPR_MD5_DIGEST_LENGTH);
```

**Buffer Overflow StrcpyStrcat\Path 7:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=7 |
| Status | New |

The size of the buffer used by ntlm_current_time in timestamp, at line 186 of FreeRDP@@FreeRDP-2.4.0-CVE-2020-11086-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ntlm_current_time passes to timestamp, at line 186 of FreeRDP@@FreeRDP-2.4.0-CVE-2020-11086-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.4.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.4.0-CVE-2020-11086-FP.c |
| Line | 186 | 193 |
| Object | timestamp | timestamp |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.4.0-CVE-2020-11086-FP.c |
| Method | void ntlm_current_time(BYTE* timestamp) |

```
....
186.  void ntlm_current_time(BYTE* timestamp)
....
193.        CopyMemory(timestamp, &(time64.QuadPart), 8);
```

**Buffer Overflow StrcpyStrcat\Path 8:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=8 |
| Status | New |

The size of the buffer used by ntlm_generate_signing_key in exported_session_key, at line 600 of FreeRDP@@FreeRDP-2.4.0-CVE-2020-11086-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ntlm_generate_signing_key passes to exported_session_key, at line 600 of FreeRDP@@FreeRDP-2.4.0-CVE-2020-11086-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.4.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.4.0-CVE-2020-11086-FP.c |
| Line | 600 | 612 |
| Object | exported_session_key | exported_session_key |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.4.0-CVE-2020-11086-FP.c |
| Method | static int ntlm_generate_signing_key(BYTE* exported_session_key, PSecBuffer sign_magic, |

```
....
600.  static int ntlm_generate_signing_key(BYTE* exported_session_key,
PSecBuffer sign_magic,
....
612.        CopyMemory(value, exported_session_key,
WINPR_MD5_DIGEST_LENGTH);
```

**Buffer Overflow StrcpyStrcat\Path 9:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=9 |
| Status | New |

The size of the buffer used by ntlm_generate_sealing_key in exported_session_key, at line 661 of FreeRDP@@FreeRDP-2.4.0-CVE-2020-11086-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ntlm_generate_sealing_key passes to exported_session_key, at line 661 of FreeRDP@@FreeRDP-2.4.0-CVE-2020-11086-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.4.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.4.0-CVE-2020-11086-FP.c |
| Line | 661 | 672 |
| Object | exported_session_key | exported_session_key |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.4.0-CVE-2020-11086-FP.c |
| Method | static int ntlm_generate_sealing_key(BYTE* exported_session_key, PSecBuffer seal_magic, |

```
....
661.  static int ntlm_generate_sealing_key(BYTE* exported_session_key,
PSecBuffer seal_magic,
....
672.        CopyMemory(p, exported_session_key,
WINPR_MD5_DIGEST_LENGTH);
```

**Buffer Overflow StrcpyStrcat\Path 10:**

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=10 |
| Status | New |

The size of the buffer used by ntlm_current_time in timestamp, at line 186 of FreeRDP@@FreeRDP-2.5.0-CVE-2020-11086-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ntlm_current_time passes to timestamp, at line 186 of FreeRDP@@FreeRDP-2.5.0-CVE-2020-11086-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.5.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.5.0-CVE-2020-11086-FP.c |
| Line | 186 | 193 |
| Object | timestamp | timestamp |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.5.0-CVE-2020-11086-FP.c |
| Method | void ntlm_current_time(BYTE* timestamp) |

```
....
186.  void ntlm_current_time(BYTE* timestamp)
....
193.        CopyMemory(timestamp, &(time64.QuadPart), 8);
```

### Buffer Overflow StrcpyStrcat\Path 11:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=11](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=11) |
| Status | New |

The size of the buffer used by ntlm_generate_signing_key in exported_session_key, at line 600 of FreeRDP@@FreeRDP-2.5.0-CVE-2020-11086-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ntlm_generate_signing_key passes to exported_session_key, at line 600 of FreeRDP@@FreeRDP-2.5.0-CVE-2020-11086-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.5.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.5.0-CVE-2020-11086-FP.c |
| Line | 600 | 612 |
| Object | exported_session_key | exported_session_key |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.5.0-CVE-2020-11086-FP.c |
| Method | static int ntlm_generate_signing_key(BYTE* exported_session_key, PSecBuffer sign_magic, |

```
....
600.  static int ntlm_generate_signing_key(BYTE* exported_session_key,
PSecBuffer sign_magic,
....
612.        CopyMemory(value, exported_session_key,
WINPR_MD5_DIGEST_LENGTH);
```

### Buffer Overflow StrcpyStrcat\Path 12:

| | |
|---|---|
| Severity | High |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=12](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=12) |
| Status | New |

The size of the buffer used by ntlm_generate_sealing_key in exported_session_key, at line 661 of FreeRDP@@FreeRDP-2.5.0-CVE-2020-11086-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ntlm_generate_sealing_key passes to exported_session_key, at line 661 of FreeRDP@@FreeRDP-2.5.0-CVE-2020-11086-FP.c, to overwrite the target buffer.

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

| | | |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.5.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.5.0-CVE-2020-11086-FP.c |
| Line | 661 | 672 |
| Object | exported_session_key | exported_session_key |

Code Snippet
File Name FreeRDP@@FreeRDP-2.5.0-CVE-2020-11086-FP.c
Method static int ntlm_generate_sealing_key(BYTE* exported_session_key, PSecBuffer seal_magic,

```
....
661.  static int ntlm_generate_sealing_key(BYTE* exported_session_key,
PSecBuffer seal_magic,
....
672.        CopyMemory(p, exported_session_key,
WINPR_MD5_DIGEST_LENGTH);
```

## 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=1000019&projectid=14&pathid=384 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 194 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-11097-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-11097-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-11097-TP.c |
| Line | 211 | 211 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2020-11097-TP.c
Method static BOOL ntlm_av_pair_add(NTLM_AV_PAIR* pAvPairList, size_t cbAvPairList, NTLM_AV_ID AvId,

```
....
211.            CopyMemory(ntlm_av_pair_get_value_pointer(pAvPair),
Value, AvLen);
```

**Dangerous Functions\Path 2:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=385 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 1126 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Line | 1167 | 1167 |
| Object | CopyMemory | CopyMemory |

| Code Snippet | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Method | SECURITY_STATUS ntlm_server_AuthenticateComplete(NTLM_CONTEXT* context) |

```
....
1167.           CopyMemory(
```

**Dangerous Functions\Path 3:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=386 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 112 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Line | 114 | 114 |
| Object | CopyMemory | CopyMemory |

| | |
|---|---|
| Code Snippet | |
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Method | static void ntlm_populate_message_header(NTLM_MESSAGE_HEADER* header, UINT32 MessageType) |

```
....
114.        CopyMemory(header->Signature, NTLM_SIGNATURE,
sizeof(NTLM_SIGNATURE));
```

**Dangerous Functions\Path 4:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=387 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 198 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Line | 268 | 268 |
| Object | CopyMemory | CopyMemory |

| | |
|---|---|
| Code Snippet | |
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Method | SECURITY_STATUS ntlm_read_NegotiateMessage(NTLM_CONTEXT* context, PSecBuffer buffer) |

```
....
268.        CopyMemory(context->NegotiateMessage.pvBuffer, buffer-
>pvBuffer, buffer->cbBuffer);
```

**Dangerous Functions\Path 5:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=388 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 285 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020- | FreeRDP@@FreeRDP-2.0.0-CVE-2020- |

| | | |
|---|---|---|
| | 13396-TP.c | 13396-TP.c |
| Line | 348 | 348 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c
Method SECURITY_STATUS ntlm_write_NegotiateMessage(NTLM_CONTEXT* context, PSecBuffer buffer)

```
....
348.         CopyMemory(context->NegotiateMessage.pvBuffer, buffer-
>pvBuffer, buffer->cbBuffer);
```

**Dangerous Functions\Path 6:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=389 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 363 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Line | 415 | 415 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c
Method SECURITY_STATUS ntlm_read_ChallengeMessage(NTLM_CONTEXT* context, PSecBuffer buffer)

```
....
415.         CopyMemory(context->ServerChallenge, message-
>ServerChallenge, 8);
```

**Dangerous Functions\Path 7:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=390 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 363 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Line | 477 | 477 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c
Method SECURITY_STATUS ntlm_read_ChallengeMessage(NTLM_CONTEXT* context, PSecBuffer buffer)

```
....
477.                CopyMemory(context->ChallengeTimestamp, ptr, 8);
```

**Dangerous Functions\Path 8:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=391 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 363 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Line | 489 | 489 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c
Method SECURITY_STATUS ntlm_read_ChallengeMessage(NTLM_CONTEXT* context, PSecBuffer buffer)

```
....
489.        CopyMemory(context->ChallengeMessage.pvBuffer, StartOffset, 
length);
```

**Dangerous Functions\Path 9:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=392 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 581 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Line | 604 | 604 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Method | SECURITY_STATUS ntlm_write_ChallengeMessage(NTLM_CONTEXT* context, PSecBuffer buffer) |

```
....
604.         CopyMemory(message->ServerChallenge, context-
>ServerChallenge, 8); /* ServerChallenge */
```

**Dangerous Functions\Path 10:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=393 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 581 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Line | 659 | 659 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Method | SECURITY_STATUS ntlm_write_ChallengeMessage(NTLM_CONTEXT* context, PSecBuffer buffer) |

```
....
659.         CopyMemory(context->ChallengeMessage.pvBuffer,
Stream_Buffer(s), length);
```

**Dangerous Functions\Path 11:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=394 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 677 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Line | 825 | 825 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Method | SECURITY_STATUS ntlm_read_AuthenticateMessage(NTLM_CONTEXT* context, PSecBuffer buffer) |

```
....
825.                CopyMemory(context->ClientChallenge, context->NTLMv2Response.Challenge.ClientChallenge, 8);
```

**Dangerous Functions\Path 12:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=395 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 677 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Line | 849 | 849 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Method | SECURITY_STATUS ntlm_read_AuthenticateMessage(NTLM_CONTEXT* context, PSecBuffer buffer) |

```
....
849.                CopyMemory(context->EncryptedRandomSessionKey,
message->EncryptedRandomSessionKey.Buffer,
```

**Dangerous Functions\Path 13:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=396](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=396) |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 677 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Line | 861 | 861 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Method | SECURITY_STATUS ntlm_read_AuthenticateMessage(NTLM_CONTEXT* context, PSecBuffer buffer) |

```
....
861.         CopyMemory(context->AuthenticateMessage.pvBuffer,
Stream_Buffer(s), length);
```

**Dangerous Functions\Path 14:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=397](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=397) |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 677 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Line | 914 | 914 |
| Object | CopyMemory | CopyMemory |

| Code Snippet | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Method | SECURITY_STATUS ntlm_read_AuthenticateMessage(NTLM_CONTEXT* context, PSecBuffer buffer) |

```
....
914.                CopyMemory(credentials->identity.User, message-
>UserName.Buffer, message->UserName.Len);
```

**Dangerous Functions\Path 15:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=398 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 677 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Line | 928 | 928 |
| Object | CopyMemory | CopyMemory |

| Code Snippet | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Method | SECURITY_STATUS ntlm_read_AuthenticateMessage(NTLM_CONTEXT* context, PSecBuffer buffer) |

```
....
928.                CopyMemory(credentials->identity.Domain, message-
>DomainName.Buffer,
```

**Dangerous Functions\Path 16:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=399 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 946 in FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

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

| | | |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Line | 1079 | 1079 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Method | SECURITY_STATUS ntlm_write_AuthenticateMessage(NTLM_CONTEXT* context, PSecBuffer buffer) |

```
....
1079.       CopyMemory(context->AuthenticateMessage.pvBuffer,
Stream_Buffer(s), length);
```

**Dangerous Functions\Path 17:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=400 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 232 in FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 257 | 257 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Method | static HRESULT STDMETHODCALLTYPE CliprdrStream_Read(IStream* This, void* pv, ULONG cb, |

```
....
257.            CopyMemory(pv, clipboard->req_fdata, clipboard-
>req_fsize);
```

**Dangerous Functions\Path 18:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=401 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 2040 in FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 2167 | 2167 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name	FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method	wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2167.                   CopyMemory(buff, globlemem, size);
```

**Dangerous Functions\Path 19:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=402](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=402) |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 2187 in FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 2218 | 2218 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name	FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method	wf_cliprdr_server_format_data_response(CliprdrClientContext* context,

```
....
2218.         CopyMemory(data, formatDataResponse->requestedFormatData,
formatDataResponse->dataLen);
```

**Dangerous Functions\Path 20:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=403](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=403) |

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

The dangerous function, CopyMemory, was found in use at line 2413 in FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 2435 | 2435 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_file_contents_response(CliprdrClientContext* context,

```
....
2435.		CopyMemory(clipboard->req_fdata, fileContentsResponse->requestedData,
```

**Dangerous Functions\Path 21:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=404 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 176 in FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c |
| Line | 211 | 211 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c
Method static BOOL nsc_rle_decompress_data(NSC_CONTEXT* context)

```
....
211.				CopyMemory(context->priv->PlaneBuffers[i], rle, originalSize);
```

**Dangerous Functions\Path 22:**

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

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=405 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 102 in FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 116 | 116 |
| Object | CopyMemory | CopyMemory |

Code Snippet

File Name FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c

Method static BOOL rdp_compute_client_auto_reconnect_cookie(rdpRdp* rdp)

```
....
116.        CopyMemory(AutoReconnectRandom, serverCookie->arcRandomBits,
16);
```

**Dangerous Functions\Path 23:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=406 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 102 in FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 120 | 120 |
| Object | CopyMemory | CopyMemory |

Code Snippet

File Name FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c

Method static BOOL rdp_compute_client_auto_reconnect_cookie(rdpRdp* rdp)

```
....
120.              CopyMemory(ClientRandom, settings->ClientRandom,
settings->ClientRandomLength);
```

**Dangerous Functions\Path 24:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=407 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 138 in FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 170 | 170 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c
Method static BOOL rdp_read_server_auto_reconnect_cookie(rdpRdp* rdp, wStream* s, logon_info_ex* info)

```
....
170.        CopyMemory(info->ArcRandomBits, p, 16);
```

**Dangerous Functions\Path 25:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=408 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 186 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 193 | 193 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c
Method void ntlm_current_time(BYTE* timestamp)

```
....
193.        CopyMemory(timestamp, &(time64.QuadPart), 8);
```

### Dangerous Functions\Path 26:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=409 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 201 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 204 | 204 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Method | void ntlm_generate_timestamp(NTLM_CONTEXT* context) |

```
....
204.            CopyMemory(context->Timestamp, context-
>ChallengeTimestamp, 8);
```

### Dangerous Functions\Path 27:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=410 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 377 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 397 | 397 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Method | int ntlm_compute_lm_v2_response(NTLM_CONTEXT* context) |

```
....
397.        CopyMemory(value, context->ServerChallenge, 8);
```

**Dangerous Functions\Path 28:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=411 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 377 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 398 | 398 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Method | int ntlm_compute_lm_v2_response(NTLM_CONTEXT* context) |

```
....
398.        CopyMemory(&value[8], context->ClientChallenge, 8);
```

**Dangerous Functions\Path 29:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=412 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 377 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 409 | 409 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c
Method int ntlm_compute_lm_v2_response(NTLM_CONTEXT* context)

```
....
409.        CopyMemory(&response[16], context->ClientChallenge, 8);
```

**Dangerous Functions\Path 30:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=413](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=413) |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 420 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 443 | 443 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c
Method int ntlm_compute_ntlm_v2_response(NTLM_CONTEXT* context)

```
....
443.        CopyMemory(&blob[8], context->Timestamp, 8);         /*
Timestamp (8 bytes) */
```

**Dangerous Functions\Path 31:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=414](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=414) |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 420 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 444 | 444 |

| | | |
|---|---|---|
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Method | int ntlm_compute_ntlm_v2_response(NTLM_CONTEXT* context) |

```
....
444.         CopyMemory(&blob[16], context->ClientChallenge, 8); /*
ClientChallenge (8 bytes) */
```

**Dangerous Functions\Path 32:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=415 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 420 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 446 | 446 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Method | int ntlm_compute_ntlm_v2_response(NTLM_CONTEXT* context) |

```
....
446.         CopyMemory(&blob[28], TargetInfo->pvBuffer, TargetInfo-
>cbBuffer);
```

**Dangerous Functions\Path 33:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=416 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 420 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020- | FreeRDP@@FreeRDP-2.2.0-CVE-2020- |

| | | |
|---|---|---|
| | 11086-FP.c | 11086-FP.c |
| Line | 458 | 458 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c
Method int ntlm_compute_ntlm_v2_response(NTLM_CONTEXT* context)

```
....
458.        CopyMemory(blob, context->ServerChallenge, 8);
```

**Dangerous Functions\Path 34:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=417 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 420 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 459 | 459 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c
Method int ntlm_compute_ntlm_v2_response(NTLM_CONTEXT* context)

```
....
459.        CopyMemory(&blob[8], ntlm_v2_temp.pvBuffer,
ntlm_v2_temp.cbBuffer);
```

**Dangerous Functions\Path 35:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=418 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 420 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 470 | 470 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c
Method int ntlm_compute_ntlm_v2_response(NTLM_CONTEXT* context)

```
....
470.        CopyMemory(blob, context->NtProofString,
WINPR_MD5_DIGEST_LENGTH);
```

**Dangerous Functions\Path 36:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=419 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 420 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 471 | 471 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c
Method int ntlm_compute_ntlm_v2_response(NTLM_CONTEXT* context)

```
....
471.        CopyMemory(&blob[16], ntlm_v2_temp.pvBuffer,
ntlm_v2_temp.cbBuffer);
```

**Dangerous Functions\Path 37:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=420 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 531 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 534 | 534 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Method | void ntlm_generate_key_exchange_key(NTLM_CONTEXT* context) |

```
....
534.        CopyMemory(context->KeyExchangeKey, context->SessionBaseKey,
16);
```

**Dangerous Functions\Path 38:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=421 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 552 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 554 | 554 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Method | void ntlm_generate_exported_session_key(NTLM_CONTEXT* context) |

```
....
554.        CopyMemory(context->ExportedSessionKey, context-
>RandomSessionKey, 16);
```

**Dangerous Functions\Path 39:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14 |

| | |
|---|---|
| | &pathid=422 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 575 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 589 | 589 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c
Method void ntlm_decrypt_random_session_key(NTLM_CONTEXT* context)

```
....
589.                CopyMemory(context->RandomSessionKey, context-
>KeyExchangeKey, 16);
```

**Dangerous Functions\Path 40:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=423 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 600 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 612 | 612 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c
Method static int ntlm_generate_signing_key(BYTE* exported_session_key, PSecBuffer sign_magic,

```
....
612.          CopyMemory(value, exported_session_key,
WINPR_MD5_DIGEST_LENGTH);
```

**Dangerous Functions\Path 41:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=424 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 600 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 613 | 613 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Method | static int ntlm_generate_signing_key(BYTE* exported_session_key, PSecBuffer sign_magic, |

```
....
613.         CopyMemory(&value[WINPR_MD5_DIGEST_LENGTH], sign_magic-
>pvBuffer, sign_magic->cbBuffer);
```

**Dangerous Functions\Path 42:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=425 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 661 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 672 | 672 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Method | static int ntlm_generate_sealing_key(BYTE* exported_session_key, PSecBuffer seal_magic, |

```
....
672.        CopyMemory(p, exported_session_key,
WINPR_MD5_DIGEST_LENGTH);
```

**Dangerous Functions\Path 43:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=426 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 661 in FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 673 | 673 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Method | static int ntlm_generate_sealing_key(BYTE* exported_session_key, PSecBuffer seal_magic, |

```
....
673.        CopyMemory(&p[WINPR_MD5_DIGEST_LENGTH], seal_magic-
>pvBuffer, seal_magic->cbBuffer);
```

**Dangerous Functions\Path 44:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=427 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 232 in FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 257 | 257 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c
Method static HRESULT STDMETHODCALLTYPE CliprdrStream_Read(IStream* This, void* pv, ULONG cb,

```
....
257.            CopyMemory(pv, clipboard->req_fdata, clipboard-
>req_fsize);
```

**Dangerous Functions\Path 45:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=428 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 2041 in FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 2168 | 2168 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2168.                   CopyMemory(buff, globlemem, size);
```

**Dangerous Functions\Path 46:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=429 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 2188 in FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |

| | | |
|---|---|---|
| Line | 2219 | 2219 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_response(CliprdrClientContext* context,

```
....
2219.       CopyMemory(data, formatDataResponse->requestedFormatData, 
formatDataResponse->dataLen);
```

**Dangerous Functions\Path 47:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=430 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 2414 in FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 2436 | 2436 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_file_contents_response(CliprdrClientContext* context,

```
....
2436.       CopyMemory(clipboard->req_fdata, fileContentsResponse-
>requestedData,
```

**Dangerous Functions\Path 48:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=431 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 176 in FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

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

| | | |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c |
| Line | 211 | 211 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c
Method static BOOL nsc_rle_decompress_data(NSC_CONTEXT* context)

```
....
211.                CopyMemory(context->priv->PlaneBuffers[i], rle, 
originalSize);
```

**Dangerous Functions\Path 49:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=432 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 150 in FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 164 | 164 |
| Object | CopyMemory | CopyMemory |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c
Method static BOOL rdp_compute_client_auto_reconnect_cookie(rdpRdp* rdp)

```
....
164.        CopyMemory(AutoReconnectRandom, serverCookie->arcRandomBits, 
16);
```

**Dangerous Functions\Path 50:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=433 |
| Status | New |

The dangerous function, CopyMemory, was found in use at line 150 in FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 168 | 168 |
| Object | CopyMemory | CopyMemory |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_compute_client_auto_reconnect_cookie(rdpRdp* rdp) |

```
....
168.                CopyMemory(ClientRandom, settings->ClientRandom,
settings->ClientRandomLength);
```

## 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=1000019&projectid=14&pathid=98 |
| Status | New |

Calling free() (line 2513) on a variable that was not dynamically allocated (line 2513) in file FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 2545 | 2545 |
| Object | clipboard | clipboard |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Method | BOOL wf_cliprdr_uninit(wfContext* wfc, CliprdrClientContext* cliprdr) |

```
....
2545.        free(clipboard);
```

**MemoryFree on StackVariable\Path 2:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=99 |
| Status | New |

Calling free() (line 441) on a variable that was not dynamically allocated (line 441) in file FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c |
| Line | 483 | 483 |
| Object | outStr | outStr |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c
Method static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp)

```
....
483.                                   free(outStr);
```

**MemoryFree on StackVariable\Path 3:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=100 |
| Status | New |

Calling free() (line 441) on a variable that was not dynamically allocated (line 441) in file FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c |
| Line | 496 | 496 |
| Object | outStr | outStr |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c
Method static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp)

```
....
496.                            free(outStr);
```

**MemoryFree on StackVariable\Path 4:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=101 |

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

Calling free() (line 441) on a variable that was not dynamically allocated (line 441) in file FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c |
| Line | 528 | 528 |
| Object | outStr | outStr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c |
| Method | static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp) |

```
....
528.                                        free(outStr);
```

**MemoryFree on StackVariable\Path 5:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=102 |
| Status | New |

Calling free() (line 441) on a variable that was not dynamically allocated (line 441) in file FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c |
| Line | 538 | 538 |
| Object | outStr | outStr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c |
| Method | static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp) |

```
....
538.                                free(outStr);
```

**MemoryFree on StackVariable\Path 6:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14 |

| | |
|---|---|
| | &pathid=103 |
| Status | New |

Calling free() (line 441) on a variable that was not dynamically allocated (line 441) in file FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c |
| Line | 483 | 483 |
| Object | outStr | outStr |

Code Snippet

File Name FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c

Method static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp)

```
....
483.                                    free(outStr);
```

**MemoryFree on StackVariable\Path 7:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=104 |
| Status | New |

Calling free() (line 441) on a variable that was not dynamically allocated (line 441) in file FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c |
| Line | 496 | 496 |
| Object | outStr | outStr |

Code Snippet

File Name FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c

Method static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp)

```
....
496.                            free(outStr);
```

**MemoryFree on StackVariable\Path 8:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=105 |
| Status | New |

Calling free() (line 441) on a variable that was not dynamically allocated (line 441) in file FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c |
| Line | 528 | 528 |
| Object | outStr | outStr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c |
| Method | static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp) |

```
....
528.                          free(outStr);
```

**MemoryFree on StackVariable\Path 9:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=106 |
| Status | New |

Calling free() (line 441) on a variable that was not dynamically allocated (line 441) in file FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c |
| Line | 538 | 538 |
| Object | outStr | outStr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c |
| Method | static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp) |

```
....
538.                     free(outStr);
```

**MemoryFree on StackVariable\Path 10:**

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

| | |
|---|---|
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=107](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=107) |
| Status | New |

Calling free() (line 138) on a variable that was not dynamically allocated (line 138) in file FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 177 | 177 |
| Object | base64 | base64 |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_read_server_auto_reconnect_cookie(rdpRdp* rdp, wStream* s, logon_info_ex* info) |

```
....
177.                  free(base64);
```

**MemoryFree on StackVariable\Path 11:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=108](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=108) |
| Status | New |

Calling free() (line 414) on a variable that was not dynamically allocated (line 414) in file FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 472 | 472 |
| Object | clientAddress | clientAddress |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_write_extended_info_packet(rdpRdp* rdp, wStream* s) |

```
....
472.            free(clientAddress);
```

**MemoryFree on StackVariable\Path 12:**

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

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=109 |
| Status | New |

Calling free() (line 414) on a variable that was not dynamically allocated (line 414) in file FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 473 | 473 |
| Object | clientDir | clientDir |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_extended_info_packet(rdpRdp* rdp, wStream* s)

```
....
473.        free(clientDir);
```

**MemoryFree on StackVariable\Path 13:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=110 |
| Status | New |

Calling free() (line 721) on a variable that was not dynamically allocated (line 721) in file FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 945 | 945 |
| Object | domainW | domainW |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
945.        free(domainW);
```

**MemoryFree on StackVariable\Path 14:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=111 |
| Status | New |

Calling free() (line 721) on a variable that was not dynamically allocated (line 721) in file FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 946 | 946 |
| Object | userNameW | userNameW |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s) |

```
....
946.         free(userNameW);
```

**MemoryFree on StackVariable\Path 15:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=112 |
| Status | New |

Calling free() (line 721) on a variable that was not dynamically allocated (line 721) in file FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 947 | 947 |
| Object | alternateShellW | alternateShellW |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s) |

```
....
947.         free(alternateShellW);
```

**MemoryFree on StackVariable\Path 16:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14 |

| | |
|---|---|
| | [&pathid=113](&pathid=113) |
| Status | New |

Calling free() (line 721) on a variable that was not dynamically allocated (line 721) in file FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 948 | 948 |
| Object | workingDirW | workingDirW |

Code Snippet

File Name FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c

Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
948.        free(workingDirW);
```

**MemoryFree on StackVariable\Path 17:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=114 |
| Status | New |

Calling free() (line 721) on a variable that was not dynamically allocated (line 721) in file FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 951 | 951 |
| Object | passwordW | passwordW |

Code Snippet

File Name FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c

Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
951.            free(passwordW);
```

**MemoryFree on StackVariable\Path 18:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=115 |

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

Calling free() (line 1377) on a variable that was not dynamically allocated (line 1377) in file FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 1397 | 1397 |
| Object | wString | wString |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_logon_info_v1(wStream* s, logon_info* info)

```
....
1397.                  free(wString);
```

**MemoryFree on StackVariable\Path 19:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=116 |
| Status | New |

Calling free() (line 1377) on a variable that was not dynamically allocated (line 1377) in file FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 1404 | 1404 |
| Object | wString | wString |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_logon_info_v1(wStream* s, logon_info* info)

```
....
1404.          free(wString);
```

**MemoryFree on StackVariable\Path 20:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=117 |
| Status | New |

Calling free() (line 1377) on a variable that was not dynamically allocated (line 1377) in file FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 1416 | 1416 |
| Object | wString | wString |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_write_logon_info_v1(wStream* s, logon_info* info) |

```
....
1416.                   free(wString);
```

**MemoryFree on StackVariable\Path 21:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=118 |
| Status | New |

Calling free() (line 1377) on a variable that was not dynamically allocated (line 1377) in file FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 1423 | 1423 |
| Object | wString | wString |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_write_logon_info_v1(wStream* s, logon_info* info) |

```
....
1423.           free(wString);
```

**MemoryFree on StackVariable\Path 22:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=119 |
| Status | New |

Calling free() (line 1429) on a variable that was not dynamically allocated (line 1429) in file FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 1457 | 1457 |
| Object | wString | wString |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_write_logon_info_v2(wStream* s, logon_info* info) |

```
....
1457.        free(wString);
```

### MemoryFree on StackVariable\Path 23:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=120 |
| Status | New |

Calling free() (line 1429) on a variable that was not dynamically allocated (line 1429) in file FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 1465 | 1465 |
| Object | wString | wString |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_write_logon_info_v2(wStream* s, logon_info* info) |

```
....
1465.        free(wString);
```

### MemoryFree on StackVariable\Path 24:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=121 |
| Status | New |

Calling free() (line 264) on a variable that was not dynamically allocated (line 264) in file FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 289 | 289 |
| Object | PasswordHash | PasswordHash |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Method | static int ntlm_convert_password_hash(NTLM_CONTEXT* context, BYTE* hash) |

```
....
289.         free(PasswordHash);
```

**MemoryFree on StackVariable\Path 25:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=122 |
| Status | New |

Calling free() (line 2514) on a variable that was not dynamically allocated (line 2514) in file FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 2546 | 2546 |
| Object | clipboard | clipboard |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | BOOL wf_cliprdr_uninit(wfContext* wfc, CliprdrClientContext* cliprdr) |

```
....
2546.        free(clipboard);
```

**MemoryFree on StackVariable\Path 26:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=123 |
| Status | New |

Calling free() (line 445) on a variable that was not dynamically allocated (line 445) in file FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c |
| Line | 487 | 487 |
| Object | outStr | outStr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c |
| Method | static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp) |

```
....
487.                                    free(outStr);
```

**MemoryFree on StackVariable\Path 27:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=124](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=124) |
| Status | New |

Calling free() (line 445) on a variable that was not dynamically allocated (line 445) in file FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c |
| Line | 500 | 500 |
| Object | outStr | outStr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c |
| Method | static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp) |

```
....
500.                            free(outStr);
```

**MemoryFree on StackVariable\Path 28:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=125](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=125) |
| Status | New |

Calling free() (line 445) on a variable that was not dynamically allocated (line 445) in file FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c |
| Line | 532 | 532 |
| Object | outStr | outStr |

Code Snippet

File Name FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c

Method static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp)

```
....
532.                                free(outStr);
```

**MemoryFree on StackVariable\Path 29:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=126 |
| Status | New |

Calling free() (line 445) on a variable that was not dynamically allocated (line 445) in file FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c |
| Line | 542 | 542 |
| Object | outStr | outStr |

Code Snippet

File Name FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c

Method static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp)

```
....
542.                          free(outStr);
```

**MemoryFree on StackVariable\Path 30:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=127 |

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

Calling free() (line 445) on a variable that was not dynamically allocated (line 445) in file FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c |
| Line | 487 | 487 |
| Object | outStr | outStr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c |
| Method | static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp) |

```
....
487.                                    free(outStr);
```

**MemoryFree on StackVariable\Path 31:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=128](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=128) |
| Status | New |

Calling free() (line 445) on a variable that was not dynamically allocated (line 445) in file FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c |
| Line | 500 | 500 |
| Object | outStr | outStr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c |
| Method | static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp) |

```
....
500.                            free(outStr);
```

**MemoryFree on StackVariable\Path 32:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14) |

| | |
|---|---|
| | &pathid=129 |
| Status | New |

Calling free() (line 445) on a variable that was not dynamically allocated (line 445) in file FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c |
| Line | 532 | 532 |
| Object | outStr | outStr |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c
Method static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp)

```
....
532.                                       free(outStr);
```

### MemoryFree on StackVariable\Path 33:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=130 |
| Status | New |

Calling free() (line 445) on a variable that was not dynamically allocated (line 445) in file FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c |
| Line | 542 | 542 |
| Object | outStr | outStr |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c
Method static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp)

```
....
542.                               free(outStr);
```

### MemoryFree on StackVariable\Path 34:

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=131 |
| Status | New |

Calling free() (line 186) on a variable that was not dynamically allocated (line 186) in file FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 225 | 225 |
| Object | base64 | base64 |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_read_server_auto_reconnect_cookie(rdpRdp* rdp, wStream* s, logon_info_ex* info) |

```
....
225.                free(base64);
```

**MemoryFree on StackVariable\Path 35:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=132 |
| Status | New |

Calling free() (line 403) on a variable that was not dynamically allocated (line 403) in file FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 461 | 461 |
| Object | clientAddress | clientAddress |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_write_extended_info_packet(rdpRdp* rdp, wStream* s) |

```
....
461.          free(clientAddress);
```

**MemoryFree on StackVariable\Path 36:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=133 |
| Status | New |

Calling free() (line 403) on a variable that was not dynamically allocated (line 403) in file FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 462 | 462 |
| Object | clientDir | clientDir |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_write_extended_info_packet(rdpRdp* rdp, wStream* s) |

```
....
462.        free(clientDir);
```

**MemoryFree on StackVariable\Path 37:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=134 |
| Status | New |

Calling free() (line 609) on a variable that was not dynamically allocated (line 609) in file FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 833 | 833 |
| Object | domainW | domainW |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s) |

```
....
833.        free(domainW);
```

**MemoryFree on StackVariable\Path 38:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14 |

| | |
|---|---|
| | &pathid=135 |
| Status | New |

Calling free() (line 609) on a variable that was not dynamically allocated (line 609) in file FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 834 | 834 |
| Object | userNameW | userNameW |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
834.		free(userNameW);
```

**MemoryFree on StackVariable\Path 39:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=136 |
| Status | New |

Calling free() (line 609) on a variable that was not dynamically allocated (line 609) in file FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 835 | 835 |
| Object | alternateShellW | alternateShellW |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
835.		free(alternateShellW);
```

**MemoryFree on StackVariable\Path 40:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=137 |

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

Calling free() (line 609) on a variable that was not dynamically allocated (line 609) in file FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 836 | 836 |
| Object | workingDirW | workingDirW |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
836.		free(workingDirW);
```

**MemoryFree on StackVariable\Path 41:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=138 |
| Status | New |

Calling free() (line 609) on a variable that was not dynamically allocated (line 609) in file FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 839 | 839 |
| Object | passwordW | passwordW |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
839.			free(passwordW);
```

**MemoryFree on StackVariable\Path 42:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=139 |
| Status | New |

Calling free() (line 1263) on a variable that was not dynamically allocated (line 1263) in file FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 1283 | 1283 |
| Object | wString | wString |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_logon_info_v1(wStream* s, logon_info* info)

```
....
1283.                  free(wString);
```

**MemoryFree on StackVariable\Path 43:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=140 |
| Status | New |

Calling free() (line 1263) on a variable that was not dynamically allocated (line 1263) in file FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 1290 | 1290 |
| Object | wString | wString |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_logon_info_v1(wStream* s, logon_info* info)

```
....
1290.          free(wString);
```

**MemoryFree on StackVariable\Path 44:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=141 |
| Status | New |

Calling free() (line 1263) on a variable that was not dynamically allocated (line 1263) in file FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 1302 | 1302 |
| Object | wString | wString |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_logon_info_v1(wStream* s, logon_info* info)

```
....
1302.                free(wString);
```

**MemoryFree on StackVariable\Path 45:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=142 |
| Status | New |

Calling free() (line 1263) on a variable that was not dynamically allocated (line 1263) in file FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 1309 | 1309 |
| Object | wString | wString |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_logon_info_v1(wStream* s, logon_info* info)

```
....
1309.          free(wString);
```

**MemoryFree on StackVariable\Path 46:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=143 |
| Status | New |

Calling free() (line 1315) on a variable that was not dynamically allocated (line 1315) in file FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 1343 | 1343 |
| Object | wString | wString |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_logon_info_v2(wStream* s, logon_info* info)

```
....
1343.        free(wString);
```

**MemoryFree on StackVariable\Path 47:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=144 |
| Status | New |

Calling free() (line 1315) on a variable that was not dynamically allocated (line 1315) in file FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 1351 | 1351 |
| Object | wString | wString |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_logon_info_v2(wStream* s, logon_info* info)

```
....
1351.        free(wString);
```

**MemoryFree on StackVariable\Path 48:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=145 |
| Status | New |

Calling free() (line 264) on a variable that was not dynamically allocated (line 264) in file FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c |
| Line | 289 | 289 |
| Object | PasswordHash | PasswordHash |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c |
| Method | static int ntlm_convert_password_hash(NTLM_CONTEXT* context, BYTE* hash) |

```
....
289.        free(PasswordHash);
```

**MemoryFree on StackVariable\Path 49:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=146 |
| Status | New |

Calling free() (line 2514) on a variable that was not dynamically allocated (line 2514) in file FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 2546 | 2546 |
| Object | clipboard | clipboard |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Method | BOOL wf_cliprdr_uninit(wfContext* wfc, CliprdrClientContext* cliprdr) |

```
....
2546.       free(clipboard);
```

**MemoryFree on StackVariable\Path 50:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=147 |
| Status | New |

Calling free() (line 445) on a variable that was not dynamically allocated (line 445) in file FreeRDP@@FreeRDP-2.3.0-CVE-2022-39347-TP.c may result with a crash.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2022-39347-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2022-39347-TP.c |
| Line | 487 | 487 |
| Object | outStr | outStr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2022-39347-TP.c |
| Method | static UINT drive_process_irp_query_volume_information(DRIVE_DEVICE* drive, IRP* irp) |

```
....
487.                              free(outStr);
```

## 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=1000019&projectid=14&pathid=916 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c |
| Line | 661 | 661 |
| Object | h264 | h264 |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c |
| Method | H264_CONTEXT* h264_context_new(BOOL Compressor) |

```
....
661.        H264_CONTEXT* h264 = (H264_CONTEXT*)calloc(1,
sizeof(H264_CONTEXT));
```

**Memory Leak\Path 2:**

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

| | |
|---|---|
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=917 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c |
| Line | 671 | 671 |
| Object | h264 | h264 |

Code Snippet

File Name FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c

Method H264_CONTEXT* h264_context_new(BOOL Compressor)

```
....
671.        H264_CONTEXT* h264 = (H264_CONTEXT*)calloc(1,
sizeof(H264_CONTEXT));
```

**Memory Leak\Path 3:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=918 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c |
| Line | 671 | 671 |
| Object | h264 | h264 |

Code Snippet

File Name FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c

Method H264_CONTEXT* h264_context_new(BOOL Compressor)

```
....
671.        H264_CONTEXT* h264 = (H264_CONTEXT*)calloc(1,
sizeof(H264_CONTEXT));
```

**Memory Leak\Path 4:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=919 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c |
| Line | 709 | 709 |
| Object | h264 | h264 |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c |
| Method | H264_CONTEXT* h264_context_new(BOOL Compressor) |

```
....
709.         H264_CONTEXT* h264 = (H264_CONTEXT*)calloc(1,
sizeof(H264_CONTEXT));
```

**Memory Leak\Path 5:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=920 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Line | 906 | 906 |
| Object | User | User |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c |
| Method | SECURITY_STATUS ntlm_read_AuthenticateMessage(NTLM_CONTEXT* context, PSecBuffer buffer) |

```
....
906.            credentials->identity.User = (UINT16*)malloc(message-
>UserName.Len);
```

**Memory Leak\Path 6:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=921 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020- | FreeRDP@@FreeRDP-2.0.0-CVE-2020- |

| | | |
|---|---|---|
| | 13396-TP.c | 13396-TP.c |
| Line | 920 | 920 |
| Object | Domain | Domain |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2020-13396-TP.c
Method SECURITY_STATUS ntlm_read_AuthenticateMessage(NTLM_CONTEXT* context, PSecBuffer buffer)

```
....
920.                credentials->identity.Domain =
(UINT16*)malloc(message->DomainName.Len);
```

**Memory Leak\Path 7:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=922 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c |
| Line | 481 | 481 |
| Object | strings | strings |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c
Method static void string_list_allocate(string_list* list, int allocate_count)

```
....
481.                list->strings = calloc((size_t)allocate_count,
sizeof(char*));
```

**Memory Leak\Path 8:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=923 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 422 | 422 |

| | | |
|---|---|---|
| Object | instance | instance |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method static CliprdrStream* CliprdrStream_New(ULONG index, void* pData, const FILEDESCRIPTORW* dsc)

```
....
422.         instance = (CliprdrStream*)calloc(1, sizeof(CliprdrStream));
```

**Memory Leak\Path 9:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=924 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 428 | 428 |
| Object | lpVtbl | lpVtbl |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method static CliprdrStream* CliprdrStream_New(ULONG index, void* pData, const FILEDESCRIPTORW* dsc)

```
....
428.              iStream->lpVtbl = (IStreamVtbl*)calloc(1,
sizeof(IStreamVtbl));
```

**Memory Leak\Path 10:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=925 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 609 | 609 |
| Object | m_pStream | m_pStream |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Method | static HRESULT STDMETHODCALLTYPE CliprdrDataObject_GetData(IDataObject* This, FORMATETC* pFormatEtc, |

```
....
609.                            instance->m_pStream =
(LPSTREAM*)calloc(instance->m_nStreams, sizeof(LPSTREAM));
```

**Memory Leak\Path 11:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=926 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 755 | 755 |
| Object | instance | instance |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Method | static CliprdrDataObject* CliprdrDataObject_New(FORMATETC* fmtetc, STGMEDIUM* stgmed, ULONG count, |

```
....
755.         instance = (CliprdrDataObject*)calloc(1,
sizeof(CliprdrDataObject));
```

**Memory Leak\Path 12:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=927 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 761 | 761 |
| Object | lpVtbl | lpVtbl |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Method | static CliprdrDataObject* CliprdrDataObject_New(FORMATETC* fmtetc, STGMEDIUM* stgmed, ULONG count, |

```
....
761.        iDataObject->lpVtbl = (IDataObjectVtbl*)calloc(1,
sizeof(IDataObjectVtbl));
```

**Memory Leak\Path 13:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=928](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=928) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 786 | 786 |
| Object | m_pFormatEtc | m_pFormatEtc |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Method | static CliprdrDataObject* CliprdrDataObject_New(FORMATETC* fmtetc, STGMEDIUM* stgmed, ULONG count, |

```
....
786.            instance->m_pFormatEtc = (FORMATETC*)calloc(count,
sizeof(FORMATETC));
```

**Memory Leak\Path 14:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=929](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=929) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 791 | 791 |
| Object | m_pStgMedium | m_pStgMedium |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Method | static CliprdrDataObject* CliprdrDataObject_New(FORMATETC* fmtetc, STGMEDIUM* stgmed, ULONG count, |

```
....
791.                instance->m_pStgMedium = (STGMEDIUM*)calloc(count,
sizeof(STGMEDIUM));
```

**Memory Leak\Path 15:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=930 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1003 | 1003 |
| Object | instance | instance |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Method | CliprdrEnumFORMATETC* CliprdrEnumFORMATETC_New(ULONG nFormats, FORMATETC* pFormatEtc) |

```
....
1003.        instance = (CliprdrEnumFORMATETC*)calloc(1,
sizeof(CliprdrEnumFORMATETC));
```

**Memory Leak\Path 16:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=931 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1009 | 1009 |
| Object | lpVtbl | lpVtbl |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Method | CliprdrEnumFORMATETC* CliprdrEnumFORMATETC_New(ULONG nFormats, FORMATETC* pFormatEtc) |

```
....
1009.       iEnumFORMATETC->lpVtbl = (IEnumFORMATETCVtbl*)calloc(1,
sizeof(IEnumFORMATETCVtbl));
```

**Memory Leak\Path 17:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=932 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1027 | 1027 |
| Object | m_pFormatEtc | m_pFormatEtc |

Code Snippet

File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c

Method CliprdrEnumFORMATETC* CliprdrEnumFORMATETC_New(ULONG nFormats, FORMATETC* pFormatEtc)

```
....
1027.           instance->m_pFormatEtc = (FORMATETC*)calloc(nFormats,
sizeof(FORMATETC));
```

**Memory Leak\Path 18:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=933 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1911 | 1911 |
| Object | name | name |

Code Snippet

File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c

Method static UINT wf_cliprdr_server_format_list(CliprdrClientContext* context,

```
....
1911.               mapping->name = calloc(size + 1, sizeof(WCHAR));
```

**Memory Leak\Path 19:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=934 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 2124 | 2124 |
| Object | wFileName | wFileName |

Code Snippet

File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c

Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2124.                    wFileName = (LPWSTR)calloc(cchWideChar,
sizeof(WCHAR));
```

**Memory Leak\Path 20:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=935 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 2455 | 2455 |
| Object | clipboard | clipboard |

Code Snippet

File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c

Method BOOL wf_cliprdr_init(wfContext* wfc, CliprdrClientContext* cliprdr)

```
....
2455.        wfc->clipboard = (wfClipboard*)calloc(1,
sizeof(wfClipboard));
```

**Memory Leak\Path 21:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=936 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 2483 | 2483 |
| Object | format_mappings | format_mappings |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Method | BOOL wf_cliprdr_init(wfContext* wfc, CliprdrClientContext* cliprdr) |

```
....
2483.       if (!(clipboard->format_mappings =
```

**Memory Leak\Path 22:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=937 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c |
| Line | 894 | 894 |
| Object | drive | drive |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2022-39347-TP.c |
| Method | static UINT drive_register_drive_path(PDEVICE_SERVICE_ENTRY_POINTS pEntryPoints, const char* name, |

```
....
894.             drive = (DRIVE_DEVICE*)calloc(1,
sizeof(DRIVE_DEVICE));
```

**Memory Leak\Path 23:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=938 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c |
| Line | 894 | 894 |
| Object | drive | drive |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2022-41877-TP.c
Method static UINT drive_register_drive_path(PDEVICE_SERVICE_ENTRY_POINTS pEntryPoints, const char* name,

```
....
894.                drive = (DRIVE_DEVICE*)calloc(1,
sizeof(DRIVE_DEVICE));
```

**Memory Leak\Path 24:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=939 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c |
| Line | 251 | 251 |
| Object | BitmapData | BitmapData |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c
Method static BOOL nsc_context_initialize(NSC_CONTEXT* context, wStream* s)

```
....
251.                context->BitmapData = calloc(1, length + 16);
```

**Memory Leak\Path 25:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=940 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c |

| | | |
|---|---|---|
| Line | 327 | 327 |
| Object | context | context |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c
Method NSC_CONTEXT* nsc_context_new(void)

```
....
327.          context = (NSC_CONTEXT*)calloc(1, sizeof(NSC_CONTEXT));
```

**Memory Leak\Path 26:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=941 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c |
| Line | 332 | 332 |
| Object | priv | priv |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c
Method NSC_CONTEXT* nsc_context_new(void)

```
....
332.          context->priv = (NSC_CONTEXT_PRIV*)calloc(1,
sizeof(NSC_CONTEXT_PRIV));
```

**Memory Leak\Path 27:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=942 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c |
| Line | 143 | 143 |
| Object | AvPairs | AvPairs |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2020-11086-FP.c

| | |
|---|---|
| Method | static int ntlm_read_ntlm_v2_client_challenge(wStream* s, NTLMv2_CLIENT_CHALLENGE* challenge) |

```
....
143.         challenge->AvPairs = (NTLM_AV_PAIR*)malloc(challenge->cbAvPairs);
```

**Memory Leak\Path 28:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=943 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 422 | 422 |
| Object | instance | instance |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | static CliprdrStream* CliprdrStream_New(ULONG index, void* pData, const FILEDESCRIPTORW* dsc) |

```
....
422.         instance = (CliprdrStream*)calloc(1, sizeof(CliprdrStream));
```

**Memory Leak\Path 29:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=944 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 428 | 428 |
| Object | lpVtbl | lpVtbl |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | static CliprdrStream* CliprdrStream_New(ULONG index, void* pData, const FILEDESCRIPTORW* dsc) |

```
....
428.                iStream->lpVtbl = (IStreamVtbl*)calloc(1,
sizeof(IStreamVtbl));
```

**Memory Leak\Path 30:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=945 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 609 | 609 |
| Object | m_pStream | m_pStream |

Code Snippet

File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c

Method static HRESULT STDMETHODCALLTYPE CliprdrDataObject_GetData(IDataObject* This, FORMATETC* pFormatEtc,

```
....
609.                            instance->m_pStream =
(LPSTREAM*)calloc(instance->m_nStreams, sizeof(LPSTREAM));
```

**Memory Leak\Path 31:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=946 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 755 | 755 |
| Object | instance | instance |

Code Snippet

File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c

Method static CliprdrDataObject* CliprdrDataObject_New(FORMATETC* fmtetc, STGMEDIUM* stgmed, ULONG count,

```
....
755.          instance = (CliprdrDataObject*)calloc(1,
sizeof(CliprdrDataObject));
```

### Memory Leak\Path 32:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=947 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 761 | 761 |
| Object | lpVtbl | lpVtbl |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | static CliprdrDataObject* CliprdrDataObject_New(FORMATETC* fmtetc, STGMEDIUM* stgmed, ULONG count, |

```
....
761.          iDataObject->lpVtbl = (IDataObjectVtbl*)calloc(1,
sizeof(IDataObjectVtbl));
```

### Memory Leak\Path 33:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=948 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 786 | 786 |
| Object | m_pFormatEtc | m_pFormatEtc |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | static CliprdrDataObject* CliprdrDataObject_New(FORMATETC* fmtetc, STGMEDIUM* stgmed, ULONG count, |

```
....
786.                instance->m_pFormatEtc = (FORMATETC*)calloc(count,
sizeof(FORMATETC));
```

**Memory Leak\Path 34:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=949 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 791 | 791 |
| Object | m_pStgMedium | m_pStgMedium |

Code Snippet

File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c

Method static CliprdrDataObject* CliprdrDataObject_New(FORMATETC* fmtetc, STGMEDIUM* stgmed, ULONG count,

```
....
791.                instance->m_pStgMedium = (STGMEDIUM*)calloc(count,
sizeof(STGMEDIUM));
```

**Memory Leak\Path 35:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=950 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1003 | 1003 |
| Object | instance | instance |

Code Snippet

File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c

Method CliprdrEnumFORMATETC* CliprdrEnumFORMATETC_New(ULONG nFormats, FORMATETC* pFormatEtc)

```
....
1003.        instance = (CliprdrEnumFORMATETC*)calloc(1,
sizeof(CliprdrEnumFORMATETC));
```

**Memory Leak\Path 36:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=951 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1009 | 1009 |
| Object | lpVtbl | lpVtbl |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | CliprdrEnumFORMATETC* CliprdrEnumFORMATETC_New(ULONG nFormats, FORMATETC* pFormatEtc) |

```
....
1009.        iEnumFORMATETC->lpVtbl = (IEnumFORMATETCVtbl*)calloc(1,
sizeof(IEnumFORMATETCVtbl));
```

**Memory Leak\Path 37:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=952 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1027 | 1027 |
| Object | m_pFormatEtc | m_pFormatEtc |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | CliprdrEnumFORMATETC* CliprdrEnumFORMATETC_New(ULONG nFormats, FORMATETC* pFormatEtc) |

```
....
1027.            instance->m_pFormatEtc = (FORMATETC*)calloc(nFormats,
sizeof(FORMATETC));
```

**Memory Leak\Path 38:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=953](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=953) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1912 | 1912 |
| Object | name | name |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | static UINT wf_cliprdr_server_format_list(CliprdrClientContext* context, |

```
....
1912.                   mapping->name = calloc(size + 1, sizeof(WCHAR));
```

**Memory Leak\Path 39:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=954](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=954) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 2125 | 2125 |
| Object | wFileName | wFileName |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | wf_cliprdr_server_format_data_request(CliprdrClientContext* context, |

```
....
2125.                       wFileName = (LPWSTR)calloc(cchWideChar,
sizeof(WCHAR));
```

### Memory Leak\Path 40:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=955 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 2456 | 2456 |
| Object | clipboard | clipboard |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c
Method BOOL wf_cliprdr_init(wfContext* wfc, CliprdrClientContext* cliprdr)

```
....
2456.        wfc->clipboard = (wfClipboard*)calloc(1,
sizeof(wfClipboard));
```

### Memory Leak\Path 41:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=956 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 2484 | 2484 |
| Object | format_mappings | format_mappings |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c
Method BOOL wf_cliprdr_init(wfContext* wfc, CliprdrClientContext* cliprdr)

```
....
2484.        if (!(clipboard->format_mappings =
```

### Memory Leak\Path 42:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=957 |

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

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c |
| Line | 898 | 898 |
| Object | drive | drive |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2022-39347-TP.c
Method static UINT drive_register_drive_path(PDEVICE_SERVICE_ENTRY_POINTS pEntryPoints, const char* name,

```
....
898.              drive = (DRIVE_DEVICE*)calloc(1,
sizeof(DRIVE_DEVICE));
```

**Memory Leak\Path 43:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=958 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c |
| Line | 898 | 898 |
| Object | drive | drive |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2022-41877-TP.c
Method static UINT drive_register_drive_path(PDEVICE_SERVICE_ENTRY_POINTS pEntryPoints, const char* name,

```
....
898.              drive = (DRIVE_DEVICE*)calloc(1,
sizeof(DRIVE_DEVICE));
```

**Memory Leak\Path 44:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=959 |
| Status | New |

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

|  |  |  |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c |
| Line | 251 | 251 |
| Object | BitmapData | BitmapData |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c
Method static BOOL nsc_context_initialize(NSC_CONTEXT* context, wStream* s)

```
....
251.                context->BitmapData = calloc(1, length + 16);
```

**Memory Leak\Path 45:**

|  |  |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=960 |
| Status | New |

|  | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c |
| Line | 327 | 327 |
| Object | context | context |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c
Method NSC_CONTEXT* nsc_context_new(void)

```
....
327.        context = (NSC_CONTEXT*)calloc(1, sizeof(NSC_CONTEXT));
```

**Memory Leak\Path 46:**

|  |  |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=961 |
| Status | New |

|  | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c |
| Line | 332 | 332 |
| Object | priv | priv |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c |
| Method | NSC_CONTEXT* nsc_context_new(void) |

```
....
332.         context->priv = (NSC_CONTEXT_PRIV*)calloc(1,
sizeof(NSC_CONTEXT_PRIV));
```

**Memory Leak\Path 47:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=962 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 106 | 106 |
| Object | ret | ret |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_read_info_null_string(UINT32 flags, wStream* s, size_t cbLen, CHAR** dst, |

```
....
106.                     ret = calloc(cbLen + 1, nullSize);
```

**Memory Leak\Path 48:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=963 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c |
| Line | 143 | 143 |
| Object | AvPairs | AvPairs |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2020-11086-FP.c |
| Method | static int ntlm_read_ntlm_v2_client_challenge(wStream* s, NTLMv2_CLIENT_CHALLENGE* challenge) |

```
....
143.        challenge->AvPairs = (NTLM_AV_PAIR*)malloc(challenge-
>cbAvPairs);
```

**Memory Leak\Path 49:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=964](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=964) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 422 | 422 |
| Object | instance | instance |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Method | static CliprdrStream* CliprdrStream_New(ULONG index, void* pData, const FILEDESCRIPTORW* dsc) |

```
....
422.        instance = (CliprdrStream*)calloc(1, sizeof(CliprdrStream));
```

**Memory Leak\Path 50:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=965](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=965) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 428 | 428 |
| Object | lpVtbl | lpVtbl |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Method | static CliprdrStream* CliprdrStream_New(ULONG index, void* pData, const FILEDESCRIPTORW* dsc) |

```
....
428.                iStream->lpVtbl = (IStreamVtbl*)calloc(1,
sizeof(IStreamVtbl));
```

## 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=1000019&projectid=14&pathid=1066 |
| Status | New |

The variable declared in formats at FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c in line 1229 is not initialized when it is used by formats at FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c in line 1229.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1237 | 1293 |
| Object | formats | formats |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method static UINT cliprdr_send_format_list(wfClipboard* clipboard)

```
....
1237.       CLIPRDR_FORMAT* formats = NULL;
....
1293.               free(formats[index].formatName);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1067 |
| Status | New |

The variable declared in formats at FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c in line 1229 is not initialized when it is used by formats at FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c in line 1229.

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

| | | |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1237 | 1293 |
| Object | formats | formats |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method static UINT cliprdr_send_format_list(wfClipboard* clipboard)

```
....
1237.       CLIPRDR_FORMAT* formats = NULL;
....
1293.               free(formats[index].formatName);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1068 |
| Status | New |

The variable declared in buff at FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c in line 2040 is not initialized when it is used by buff at FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c in line 2040.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 2045 | 2175 |
| Object | buff | buff |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2045.       void* buff = NULL;
....
2175.       response.requestedFormatData = (BYTE*)buff;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1069 |
| Status | New |

The variable declared in andBits at FreeRDP@@FreeRDP-2.0.0-CVE-2024-32659-TP.c in line 305 is not initialized when it is used by andBits at FreeRDP@@FreeRDP-2.0.0-CVE-2024-32659-TP.c in line 305.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32659-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32659-TP.c |
| Line | 438 | 495 |
| Object | andBits | andBits |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2024-32659-TP.c
Method BOOL freerdp_image_copy_from_pointer_data(BYTE* pDstData, UINT32 DstFormat, UINT32 nDstStep,

```
....
438.                          const BYTE* andBits = NULL;
....
495.                                            andBits++;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1070 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.2.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.2.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-13397-FP.c |
| Line | 43 | 93 |
| Object | agreedFormat | agreedFormat |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2020-13397-FP.c
Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
43.   AUDIO_FORMAT* agreedFormat = NULL;
....
93.   recorderState.dataFormat.mSampleRate = agreedFormat-
>nSamplesPerSec;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1071 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.2.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.2.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-13397-FP.c |
| Line | 43 | 99 |
| Object | agreedFormat | agreedFormat |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2020-13397-FP.c |
| Method | static void mf_peer_rdpsnd_activated(RdpsndServerContext* context) |

```
....
43.    AUDIO_FORMAT* agreedFormat = NULL;
....
99.    recorderState.dataFormat.mChannelsPerFrame = agreedFormat-
>nChannels;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1072 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.2.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.2.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2020-13397-FP.c |
| Line | 43 | 100 |
| Object | agreedFormat | agreedFormat |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2020-13397-FP.c |
| Method | static void mf_peer_rdpsnd_activated(RdpsndServerContext* context) |

```
....
43.    AUDIO_FORMAT* agreedFormat = NULL;
....
100.        recorderState.dataFormat.mBitsPerChannel = agreedFormat-
>wBitsPerSample;
```

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

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

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1073 |
| Status | New |

The variable declared in formats at FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c in line 1229 is not initialized when it is used by formats at FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c in line 1229.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1237 | 1294 |
| Object | formats | formats |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | static UINT cliprdr_send_format_list(wfClipboard* clipboard) |

```
....
1237.        CLIPRDR_FORMAT* formats = NULL;
....
1294.                free(formats[index].formatName);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1074 |
| Status | New |

The variable declared in formats at FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c in line 1229 is not initialized when it is used by formats at FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c in line 1229.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1237 | 1294 |
| Object | formats | formats |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | static UINT cliprdr_send_format_list(wfClipboard* clipboard) |

```
....
1237.        CLIPRDR_FORMAT* formats = NULL;
....
1294.                free(formats[index].formatName);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1075 |
| Status | New |

The variable declared in buff at FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c in line 2041 is not initialized when it is used by buff at FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c in line 2041.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 2046 | 2176 |
| Object | buff | buff |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | wf_cliprdr_server_format_data_request(CliprdrClientContext* context, |

```
....
2046.        void* buff = NULL;
....
2176.        response.requestedFormatData = (BYTE*)buff;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1076 |
| Status | New |

The variable declared in andBits at FreeRDP@@FreeRDP-2.2.0-CVE-2024-32659-TP.c in line 390 is not initialized when it is used by andBits at FreeRDP@@FreeRDP-2.2.0-CVE-2024-32659-TP.c in line 390.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32659-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32659-TP.c |
| Line | 440 | 496 |
| Object | andBits | andBits |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32659-TP.c |
| Method | static BOOL freerdp_image_copy_from_pointer_data_xbpp(BYTE* pDstData, UINT32 DstFormat, |

```
....
440.                const BYTE* andBits = NULL;
....
496.                                    andBits++;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1077 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.3.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.3.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2020-13397-FP.c |
| Line | 43 | 93 |
| Object | agreedFormat | agreedFormat |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2020-13397-FP.c |
| Method | static void mf_peer_rdpsnd_activated(RdpsndServerContext* context) |

```
....
43.   AUDIO_FORMAT* agreedFormat = NULL;
....
93.   recorderState.dataFormat.mSampleRate = agreedFormat-
>nSamplesPerSec;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1078 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.3.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.3.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2020-13397-FP.c |
| Line | 43 | 99 |
| Object | agreedFormat | agreedFormat |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2020-13397-FP.c |
| Method | static void mf_peer_rdpsnd_activated(RdpsndServerContext* context) |

```
....
43.    AUDIO_FORMAT* agreedFormat = NULL;
....
99.    recorderState.dataFormat.mChannelsPerFrame = agreedFormat-
>nChannels;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1079 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.3.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.3.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2020-13397-FP.c |
| Line | 43 | 100 |
| Object | agreedFormat | agreedFormat |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2020-13397-FP.c |
| Method | static void mf_peer_rdpsnd_activated(RdpsndServerContext* context) |

```
....
43.    AUDIO_FORMAT* agreedFormat = NULL;
....
100.          recorderState.dataFormat.mBitsPerChannel = agreedFormat-
>wBitsPerSample;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1080 |
| Status | New |

The variable declared in formats at FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c in line 1229 is not initialized when it is used by formats at FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c in line 1229.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |

| | | |
|---|---|---|
| Line | 1237 | 1294 |
| Object | formats | formats |

Code Snippet
File Name FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c
Method static UINT cliprdr_send_format_list(wfClipboard* clipboard)

```
....
1237.       CLIPRDR_FORMAT* formats = NULL;
....
1294.             free(formats[index].formatName);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1081](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1081) |
| Status | New |

The variable declared in formats at FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c in line 1229 is not initialized when it is used by formats at FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c in line 1229.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 1237 | 1294 |
| Object | formats | formats |

Code Snippet
File Name FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c
Method static UINT cliprdr_send_format_list(wfClipboard* clipboard)

```
....
1237.       CLIPRDR_FORMAT* formats = NULL;
....
1294.             free(formats[index].formatName);
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1082](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1082) |
| Status | New |

The variable declared in buff at FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c in line 2041 is not initialized when it is used by buff at FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c in line 2041.

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

| | | |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 2046 | 2176 |
| Object | buff | buff |

Code Snippet
File Name FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2046.        void* buff = NULL;
....
2176.        response.requestedFormatData = (BYTE*)buff;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1083 |
| Status | New |

The variable declared in andBits at FreeRDP@@FreeRDP-2.3.0-CVE-2024-32659-TP.c in line 394 is not initialized when it is used by andBits at FreeRDP@@FreeRDP-2.3.0-CVE-2024-32659-TP.c in line 394.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2024-32659-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2024-32659-TP.c |
| Line | 444 | 500 |
| Object | andBits | andBits |

Code Snippet
File Name FreeRDP@@FreeRDP-2.3.0-CVE-2024-32659-TP.c
Method static BOOL freerdp_image_copy_from_pointer_data_xbpp(BYTE* pDstData, UINT32 DstFormat,

```
....
444.            const BYTE* andBits = NULL;
....
500.                        andBits++;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1084 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.4.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.4.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.4.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.4.0-CVE-2020-13397-FP.c |
| Line | 43 | 93 |
| Object | agreedFormat | agreedFormat |

Code Snippet
File Name FreeRDP@@FreeRDP-2.4.0-CVE-2020-13397-FP.c
Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
43.    AUDIO_FORMAT* agreedFormat = NULL;
....
93.    recorderState.dataFormat.mSampleRate = agreedFormat-
>nSamplesPerSec;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1085 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.4.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.4.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.4.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.4.0-CVE-2020-13397-FP.c |
| Line | 43 | 99 |
| Object | agreedFormat | agreedFormat |

Code Snippet
File Name FreeRDP@@FreeRDP-2.4.0-CVE-2020-13397-FP.c
Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
43.    AUDIO_FORMAT* agreedFormat = NULL;
....
99.    recorderState.dataFormat.mChannelsPerFrame = agreedFormat-
>nChannels;
```

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

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1086 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.4.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.4.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.4.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.4.0-CVE-2020-13397-FP.c |
| Line | 43 | 100 |
| Object | agreedFormat | agreedFormat |

Code Snippet
File Name FreeRDP@@FreeRDP-2.4.0-CVE-2020-13397-FP.c
Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
43.   AUDIO_FORMAT* agreedFormat = NULL;
....
100.          recorderState.dataFormat.mBitsPerChannel = agreedFormat->wBitsPerSample;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1087 |
| Status | New |

The variable declared in andBits at FreeRDP@@FreeRDP-2.4.0-CVE-2024-32659-TP.c in line 394 is not initialized when it is used by andBits at FreeRDP@@FreeRDP-2.4.0-CVE-2024-32659-TP.c in line 394.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.4.0-CVE-2024-32659-TP.c | FreeRDP@@FreeRDP-2.4.0-CVE-2024-32659-TP.c |
| Line | 444 | 500 |
| Object | andBits | andBits |

Code Snippet
File Name FreeRDP@@FreeRDP-2.4.0-CVE-2024-32659-TP.c
Method static BOOL freerdp_image_copy_from_pointer_data_xbpp(BYTE* pDstData, UINT32 DstFormat,

```
....
444.            const BYTE* andBits = NULL;
....
500.                              andBits++;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1088 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.5.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.5.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.5.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.5.0-CVE-2020-13397-FP.c |
| Line | 43 | 93 |
| Object | agreedFormat | agreedFormat |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.5.0-CVE-2020-13397-FP.c |
| Method | static void mf_peer_rdpsnd_activated(RdpsndServerContext* context) |

```
....
43.    AUDIO_FORMAT* agreedFormat = NULL;
....
93.    recorderState.dataFormat.mSampleRate = agreedFormat-
>nSamplesPerSec;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1089 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.5.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.5.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.5.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.5.0-CVE-2020-13397-FP.c |
| Line | 43 | 99 |
| Object | agreedFormat | agreedFormat |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.5.0-CVE-2020-13397-FP.c |
| Method | static void mf_peer_rdpsnd_activated(RdpsndServerContext* context) |

```
....
43.   AUDIO_FORMAT* agreedFormat = NULL;
....
99.   recorderState.dataFormat.mChannelsPerFrame = agreedFormat-
>nChannels;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1090 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.5.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.5.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.5.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.5.0-CVE-2020-13397-FP.c |
| Line | 43 | 100 |
| Object | agreedFormat | agreedFormat |

Code Snippet
File Name FreeRDP@@FreeRDP-2.5.0-CVE-2020-13397-FP.c
Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
43.   AUDIO_FORMAT* agreedFormat = NULL;
....
100.        recorderState.dataFormat.mBitsPerChannel = agreedFormat-
>wBitsPerSample;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1091 |
| Status | New |

The variable declared in andBits at FreeRDP@@FreeRDP-2.5.0-CVE-2024-32659-TP.c in line 394 is not initialized when it is used by andBits at FreeRDP@@FreeRDP-2.5.0-CVE-2024-32659-TP.c in line 394.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.5.0-CVE-2024-32659-TP.c | FreeRDP@@FreeRDP-2.5.0-CVE-2024-32659-TP.c |
| Line | 444 | 500 |
| Object | andBits | andBits |

Code Snippet
File Name FreeRDP@@FreeRDP-2.5.0-CVE-2024-32659-TP.c
Method static BOOL freerdp_image_copy_from_pointer_data_xbpp(BYTE* pDstData, UINT32 DstFormat,

```
....
444.                const BYTE* andBits = NULL;
....
500.                                andBits++;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1092 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.7.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.7.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.7.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.7.0-CVE-2020-13397-FP.c |
| Line | 43 | 93 |
| Object | agreedFormat | agreedFormat |

Code Snippet
File Name FreeRDP@@FreeRDP-2.7.0-CVE-2020-13397-FP.c
Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
43.    AUDIO_FORMAT* agreedFormat = NULL;
....
93.    recorderState.dataFormat.mSampleRate = agreedFormat-
>nSamplesPerSec;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1093 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.7.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.7.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.7.0-CVE-2020- | FreeRDP@@FreeRDP-2.7.0-CVE-2020- |

| | | |
|---|---|---|
| | 13397-FP.c | 13397-FP.c |
| Line | 43 | 99 |
| Object | agreedFormat | agreedFormat |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.7.0-CVE-2020-13397-FP.c |
| Method | static void mf_peer_rdpsnd_activated(RdpsndServerContext* context) |

```
....
43.    AUDIO_FORMAT* agreedFormat = NULL;
....
99.    recorderState.dataFormat.mChannelsPerFrame = agreedFormat-
>nChannels;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1094 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.7.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.7.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.7.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.7.0-CVE-2020-13397-FP.c |
| Line | 43 | 100 |
| Object | agreedFormat | agreedFormat |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.7.0-CVE-2020-13397-FP.c |
| Method | static void mf_peer_rdpsnd_activated(RdpsndServerContext* context) |

```
....
43.    AUDIO_FORMAT* agreedFormat = NULL;
....
100.         recorderState.dataFormat.mBitsPerChannel = agreedFormat-
>wBitsPerSample;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1095 |
| Status | New |

The variable declared in andBits at FreeRDP@@FreeRDP-2.7.0-CVE-2024-32659-TP.c in line 394 is not initialized when it is used by andBits at FreeRDP@@FreeRDP-2.7.0-CVE-2024-32659-TP.c in line 394.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.7.0-CVE-2024-32659-TP.c | FreeRDP@@FreeRDP-2.7.0-CVE-2024-32659-TP.c |
| Line | 444 | 500 |
| Object | andBits | andBits |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.7.0-CVE-2024-32659-TP.c |
| Method | static BOOL freerdp_image_copy_from_pointer_data_xbpp(BYTE* pDstData, UINT32 DstFormat, |

```
....
444.                const BYTE* andBits = NULL;
....
500.                                    andBits++;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1096 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.8.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.8.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.8.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.8.0-CVE-2020-13397-FP.c |
| Line | 43 | 93 |
| Object | agreedFormat | agreedFormat |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.8.0-CVE-2020-13397-FP.c |
| Method | static void mf_peer_rdpsnd_activated(RdpsndServerContext* context) |

```
....
43.    AUDIO_FORMAT* agreedFormat = NULL;
....
93.    recorderState.dataFormat.mSampleRate = agreedFormat-
>nSamplesPerSec;
```

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

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1097 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.8.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.8.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.8.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.8.0-CVE-2020-13397-FP.c |
| Line | 43 | 99 |
| Object | agreedFormat | agreedFormat |

Code Snippet

File Name FreeRDP@@FreeRDP-2.8.0-CVE-2020-13397-FP.c

Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
43.    AUDIO_FORMAT* agreedFormat = NULL;
....
99.    recorderState.dataFormat.mChannelsPerFrame = agreedFormat-
>nChannels;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1098 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.8.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.8.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.8.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.8.0-CVE-2020-13397-FP.c |
| Line | 43 | 100 |
| Object | agreedFormat | agreedFormat |

Code Snippet

File Name FreeRDP@@FreeRDP-2.8.0-CVE-2020-13397-FP.c

Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
43.    AUDIO_FORMAT* agreedFormat = NULL;
....
100.          recorderState.dataFormat.mBitsPerChannel = agreedFormat-
>wBitsPerSample;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1099](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1099) |
| Status | New |

The variable declared in andBits at FreeRDP@@FreeRDP-2.8.0-CVE-2024-32659-TP.c in line 394 is not initialized when it is used by andBits at FreeRDP@@FreeRDP-2.8.0-CVE-2024-32659-TP.c in line 394.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.8.0-CVE-2024-32659-TP.c | FreeRDP@@FreeRDP-2.8.0-CVE-2024-32659-TP.c |
| Line | 444 | 500 |
| Object | andBits | andBits |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.8.0-CVE-2024-32659-TP.c |
| Method | static BOOL freerdp_image_copy_from_pointer_data_xbpp(BYTE* pDstData, UINT32 DstFormat, |

```
....
444.                const BYTE* andBits = NULL;
....
500.                                  andBits++;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1100](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1100) |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.9.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.9.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.9.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.9.0-CVE-2020-13397-FP.c |
| Line | 43 | 93 |
| Object | agreedFormat | agreedFormat |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.9.0-CVE-2020-13397-FP.c |
| Method | static void mf_peer_rdpsnd_activated(RdpsndServerContext* context) |

```
....
43.   AUDIO_FORMAT* agreedFormat = NULL;
....
93.   recorderState.dataFormat.mSampleRate = agreedFormat-
>nSamplesPerSec;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1101 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.9.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.9.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.9.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.9.0-CVE-2020-13397-FP.c |
| Line | 43 | 99 |
| Object | agreedFormat | agreedFormat |

Code Snippet
File Name FreeRDP@@FreeRDP-2.9.0-CVE-2020-13397-FP.c
Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
43.   AUDIO_FORMAT* agreedFormat = NULL;
....
99.   recorderState.dataFormat.mChannelsPerFrame = agreedFormat-
>nChannels;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1102 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-2.9.0-CVE-2020-13397-FP.c in line 38 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-2.9.0-CVE-2020-13397-FP.c in line 38.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.9.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-2.9.0-CVE-2020-13397-FP.c |
| Line | 43 | 100 |
| Object | agreedFormat | agreedFormat |

| | |
|---|---|
| Code Snippet | |
| File Name | FreeRDP@@FreeRDP-2.9.0-CVE-2020-13397-FP.c |
| Method | static void mf_peer_rdpsnd_activated(RdpsndServerContext* context) |

```
....
43.   AUDIO_FORMAT* agreedFormat = NULL;
....
100.          recorderState.dataFormat.mBitsPerChannel = agreedFormat-
>wBitsPerSample;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1103 |
| Status | New |

The variable declared in andBits at FreeRDP@@FreeRDP-2.9.0-CVE-2024-32659-TP.c in line 394 is not initialized when it is used by andBits at FreeRDP@@FreeRDP-2.9.0-CVE-2024-32659-TP.c in line 394.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.9.0-CVE-2024-32659-TP.c | FreeRDP@@FreeRDP-2.9.0-CVE-2024-32659-TP.c |
| Line | 444 | 500 |
| Object | andBits | andBits |

| | |
|---|---|
| Code Snippet | |
| File Name | FreeRDP@@FreeRDP-2.9.0-CVE-2024-32659-TP.c |
| Method | static BOOL freerdp_image_copy_from_pointer_data_xbpp(BYTE* pDstData, UINT32 DstFormat, |

```
....
444.                const BYTE* andBits = NULL;
....
500.                                  andBits++;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1104 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2020-13397-FP.c in line 37 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2020-13397-FP.c in line 37.

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

| | | |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2020-13397-FP.c |
| Line | 42 | 92 |
| Object | agreedFormat | agreedFormat |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2020-13397-FP.c
Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
42.    AUDIO_FORMAT* agreedFormat = NULL;
....
92.    recorderState.dataFormat.mSampleRate = agreedFormat-
>nSamplesPerSec;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1105 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2020-13397-FP.c in line 37 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2020-13397-FP.c in line 37.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2020-13397-FP.c |
| Line | 42 | 98 |
| Object | agreedFormat | agreedFormat |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2020-13397-FP.c
Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
42.    AUDIO_FORMAT* agreedFormat = NULL;
....
98.    recorderState.dataFormat.mChannelsPerFrame = agreedFormat-
>nChannels;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1106 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2020-13397-FP.c in line 37 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2020-13397-FP.c in line 37.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2020-13397-FP.c |
| Line | 42 | 99 |
| Object | agreedFormat | agreedFormat |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2020-13397-FP.c
Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
42.   AUDIO_FORMAT* agreedFormat = NULL;
....
99.   recorderState.dataFormat.mBitsPerChannel = agreedFormat-
>wBitsPerSample;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1107 |
| Status | New |

The variable declared in andBits at FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32659-TP.c in line 391 is not initialized when it is used by andBits at FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32659-TP.c in line 391.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32659-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32659-TP.c |
| Line | 439 | 495 |
| Object | andBits | andBits |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32659-TP.c
Method static BOOL freerdp_image_copy_from_pointer_data_xbpp(BYTE* pDstData, UINT32 DstFormat,

```
....
439.              const BYTE* andBits = NULL;
....
495.                                  andBits++;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1108](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1108) |
| Status | New |

The variable declared in data at FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c in line 163 is not initialized when it is used by data at FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c in line 163.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Line | 166 | 171 |
| Object | data | data |

Code Snippet

File Name FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c

Method static BOOL rdp_redirection_read_unicode_string(wStream* s, char** str, size_t maxLength)

```
....
166.        const BYTE* data = NULL;
....
171.        const WCHAR* wstr = (const WCHAR*)data;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1109](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1109) |
| Status | New |

The variable declared in ptr at FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c in line 243 is not initialized when it is used by ptr at FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c in line 243.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Line | 248 | 252 |
| Object | ptr | ptr |

Code Snippet

File Name FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c

Method static BOOL rdp_redirection_read_base64_wchar(UINT32 flag, wStream* s, UINT32* pLength,

```
....
248.        const BYTE* ptr = NULL;
....
252.        const WCHAR* wchar = (const WCHAR*)ptr;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1110 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2020-13397-FP.c in line 37 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2020-13397-FP.c in line 37.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2020-13397-FP.c |
| Line | 42 | 92 |
| Object | agreedFormat | agreedFormat |

Code Snippet

File Name FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2020-13397-FP.c

Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
42.    AUDIO_FORMAT* agreedFormat = NULL;
....
92.    recorderState.dataFormat.mSampleRate = agreedFormat-
>nSamplesPerSec;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1111 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2020-13397-FP.c in line 37 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2020-13397-FP.c in line 37.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2020-13397-FP.c |
| Line | 42 | 98 |

| | | |
|---|---|---|
| Object | agreedFormat | agreedFormat |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2020-13397-FP.c
Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
42.    AUDIO_FORMAT* agreedFormat = NULL;
....
98.    recorderState.dataFormat.mChannelsPerFrame = agreedFormat-
>nChannels;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1112 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2020-13397-FP.c in line 37 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2020-13397-FP.c in line 37.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2020-13397-FP.c |
| Line | 42 | 99 |
| Object | agreedFormat | agreedFormat |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2020-13397-FP.c
Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
42.    AUDIO_FORMAT* agreedFormat = NULL;
....
99.    recorderState.dataFormat.mBitsPerChannel = agreedFormat-
>wBitsPerSample;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1113 |
| Status | New |

The variable declared in data at FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c in line 165 is not initialized when it is used by data at FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c in line 165.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Line | 168 | 173 |
| Object | data | data |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c
Method static BOOL rdp_redirection_read_unicode_string(wStream* s, char** str, size_t maxLength)

```
....
168.          const BYTE* data = NULL;
....
173.          const WCHAR* wstr = (const WCHAR*)data;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1114 |
| Status | New |

The variable declared in ptr at FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c in line 245 is not initialized when it is used by ptr at FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c in line 245.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Line | 250 | 254 |
| Object | ptr | ptr |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c
Method static BOOL rdp_redirection_read_base64_wchar(UINT32 flag, wStream* s, UINT32* pLength,

```
....
250.          const BYTE* ptr = NULL;
....
254.          const WCHAR* wchar = (const WCHAR*)ptr;
```

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

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1115 |
| Status | New |

The variable declared in agreedFormat at FreeRDP@@FreeRDP-3.4.0-CVE-2020-13397-FP.c in line 37 is not initialized when it is used by agreedFormat at FreeRDP@@FreeRDP-3.4.0-CVE-2020-13397-FP.c in line 37.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.4.0-CVE-2020-13397-FP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2020-13397-FP.c |
| Line | 41 | 92 |
| Object | agreedFormat | agreedFormat |

Code Snippet
File Name FreeRDP@@FreeRDP-3.4.0-CVE-2020-13397-FP.c
Method static void mf_peer_rdpsnd_activated(RdpsndServerContext* context)

```
....
41.    AUDIO_FORMAT* agreedFormat = NULL;
....
92.    recorderState.dataFormat.mSampleRate = agreedFormat-
>nSamplesPerSec;
```

## 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=1000019&projectid=14&pathid=16 |
| Status | New |

The size of the buffer used by stun_parse_attr_error_code in uint32_t, at line 235 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_attr_error_code passes to uint32_t, at line 235 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 240 | 240 |
| Object | uint32_t | uint32_t |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c

| | |
|---|---|
| Method | int stun_parse_attr_error_code(stun_attr_t *attr, const unsigned char *p, unsigned len) { |

```
....
240.     memcpy(&tmp, p, sizeof(uint32_t));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 2:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=17 |
| Status | New |

The size of the buffer used by stun_parse_attr_uint32 in uint32_t, at line 257 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_attr_uint32 passes to uint32_t, at line 257 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 262 | 262 |
| Object | uint32_t | uint32_t |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Method | int stun_parse_attr_uint32(stun_attr_t *attr, const unsigned char *p, unsigned len) |

```
....
262.     memcpy(&tmp, p, sizeof(uint32_t));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 3:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=18 |
| Status | New |

The size of the buffer used by stun_encode_address in tmp, at line 355 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_encode_address passes to tmp, at line 355 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 366 | 366 |
| Object | tmp | tmp |

| | |
|---|---|
| Code Snippet | |
| File Name | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Method | int stun_encode_address(stun_attr_t *attr) { |

```
....
366.    memcpy(attr->enc_buf.data+4, &tmp, sizeof(tmp));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 4:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=19 |
| Status | New |

The size of the buffer used by stun_parse_attr_error_code in uint32_t, at line 235 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_attr_error_code passes to uint32_t, at line 235 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 240 | 240 |
| Object | uint32_t | uint32_t |

| | |
|---|---|
| Code Snippet | |
| File Name | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Method | int stun_parse_attr_error_code(stun_attr_t *attr, const unsigned char *p, unsigned len) { |

```
....
240.    memcpy(&tmp, p, sizeof(uint32_t));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 5:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=20 |
| Status | New |

The size of the buffer used by stun_parse_attr_uint32 in uint32_t, at line 257 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_attr_uint32 passes to uint32_t, at line 257 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |

| | | |
|---|---|---|
| Line | 262 | 262 |
| Object | uint32_t | uint32_t |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Method | int stun_parse_attr_uint32(stun_attr_t *attr, const unsigned char *p, unsigned len) |

```
....
262.    memcpy(&tmp, p, sizeof(uint32_t));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 6:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=21 |
| Status | New |

The size of the buffer used by stun_encode_address in tmp, at line 355 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_encode_address passes to tmp, at line 355 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 366 | 366 |
| Object | tmp | tmp |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Method | int stun_encode_address(stun_attr_t *attr) { |

```
....
366.    memcpy(attr->enc_buf.data+4, &tmp, sizeof(tmp));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 7:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=22 |
| Status | New |

The size of the buffer used by stun_parse_attr_error_code in uint32_t, at line 235 of freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_attr_error_code passes to uint32_t, at line 235 of freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c, to overwrite the target buffer.

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

| | | |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 240 | 240 |
| Object | uint32_t | uint32_t |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c
Method int stun_parse_attr_error_code(stun_attr_t *attr, const unsigned char *p, unsigned len) {

```
....
240.    memcpy(&tmp, p, sizeof(uint32_t));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 8:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=23 |
| Status | New |

The size of the buffer used by stun_parse_attr_uint32 in uint32_t, at line 257 of freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_attr_uint32 passes to uint32_t, at line 257 of freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 262 | 262 |
| Object | uint32_t | uint32_t |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c
Method int stun_parse_attr_uint32(stun_attr_t *attr, const unsigned char *p, unsigned len)

```
....
262.    memcpy(&tmp, p, sizeof(uint32_t));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 9:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=24 |
| Status | New |

The size of the buffer used by stun_encode_address in tmp, at line 355 of freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_encode_address passes to tmp, at line 355 of freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 366 | 366 |
| Object | tmp | tmp |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c
Method int stun_encode_address(stun_attr_t *attr) {

```
....
366.    memcpy(attr->enc_buf.data+4, &tmp, sizeof(tmp));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 10:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=25 |
| Status | New |

The size of the buffer used by stun_parse_attr_error_code in uint32_t, at line 235 of freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_attr_error_code passes to uint32_t, at line 235 of freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c |
| Line | 240 | 240 |
| Object | uint32_t | uint32_t |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c
Method int stun_parse_attr_error_code(stun_attr_t *attr, const unsigned char *p, unsigned len) {

```
....
240.    memcpy(&tmp, p, sizeof(uint32_t));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 11:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=26 |
| Status | New |

The size of the buffer used by stun_parse_attr_uint32 in uint32_t, at line 257 of freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_attr_uint32 passes to uint32_t, at line 257 of freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c |
| Line | 262 | 262 |
| Object | uint32_t | uint32_t |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c
Method int stun_parse_attr_uint32(stun_attr_t *attr, const unsigned char *p, unsigned len)

```
....
262.    memcpy(&tmp, p, sizeof(uint32_t));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 12:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=27 |
| Status | New |

The size of the buffer used by stun_encode_address in tmp, at line 355 of freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_encode_address passes to tmp, at line 355 of freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c |
| Line | 366 | 366 |
| Object | tmp | tmp |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c
Method int stun_encode_address(stun_attr_t *attr) {

```
....
366.    memcpy(attr->enc_buf.data+4, &tmp, sizeof(tmp));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 13:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=28 |

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

The size of the buffer used by stun_parse_attr_error_code in uint32_t, at line 235 of freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_attr_error_code passes to uint32_t, at line 235 of freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c |
| Line | 240 | 240 |
| Object | uint32_t | uint32_t |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c |
| Method | int stun_parse_attr_error_code(stun_attr_t *attr, const unsigned char *p, unsigned len) { |

```
....
240.     memcpy(&tmp, p, sizeof(uint32_t));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 14:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=29 |
| Status | New |

The size of the buffer used by stun_parse_attr_uint32 in uint32_t, at line 257 of freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_attr_uint32 passes to uint32_t, at line 257 of freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c |
| Line | 262 | 262 |
| Object | uint32_t | uint32_t |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c |
| Method | int stun_parse_attr_uint32(stun_attr_t *attr, const unsigned char *p, unsigned len) |

```
....
262.     memcpy(&tmp, p, sizeof(uint32_t));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 15:**

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

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=30 |
| Status | New |

The size of the buffer used by stun_encode_address in tmp, at line 355 of freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_encode_address passes to tmp, at line 355 of freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c |
| Line | 366 | 366 |
| Object | tmp | tmp |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c
Method int stun_encode_address(stun_attr_t *attr) {

```
....
366.     memcpy(attr->enc_buf.data+4, &tmp, sizeof(tmp));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 16:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=31 |
| Status | New |

The size of the buffer used by parsing_error in Namespace1150220909, at line 1913 of freeswitch@@sofia-sip-v1.13.2-CVE-2022-31003-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parsing_error passes to Namespace1150220909, at line 1913 of freeswitch@@sofia-sip-v1.13.2-CVE-2022-31003-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2022-31003-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2022-31003-TP.c |
| Line | 1918 | 1918 |
| Object | Namespace1150220909 | Namespace1150220909 |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2022-31003-TP.c
Method static int parsing_error(sdp_parser_t *p, char const *fmt, ...)

```
....
1918.     memset(p->pr_error, 0, sizeof(p->pr_error));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 17:**

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

| | |
|---|---|
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=32 |
| Status | New |

The size of the buffer used by parsing_error in Namespace1716365138, at line 1913 of freeswitch@@sofia-sip-v1.13.3-CVE-2022-31003-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parsing_error passes to Namespace1716365138, at line 1913 of freeswitch@@sofia-sip-v1.13.3-CVE-2022-31003-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.3-CVE-2022-31003-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2022-31003-TP.c |
| Line | 1918 | 1918 |
| Object | Namespace1716365138 | Namespace1716365138 |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.3-CVE-2022-31003-TP.c
Method static int parsing_error(sdp_parser_t *p, char const *fmt, ...)

```
....
1918.     memset(p->pr_error, 0, sizeof(p->pr_error));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 18:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=33 |
| Status | New |

The size of the buffer used by parsing_error in Namespace294899895, at line 1913 of freeswitch@@sofia-sip-v1.13.4-CVE-2022-31003-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parsing_error passes to Namespace294899895, at line 1913 of freeswitch@@sofia-sip-v1.13.4-CVE-2022-31003-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.4-CVE-2022-31003-TP.c | freeswitch@@sofia-sip-v1.13.4-CVE-2022-31003-TP.c |
| Line | 1918 | 1918 |
| Object | Namespace294899895 | Namespace294899895 |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.4-CVE-2022-31003-TP.c
Method static int parsing_error(sdp_parser_t *p, char const *fmt, ...)

```
....
1918.     memset(p->pr_error, 0, sizeof(p->pr_error));
```

**Buffer Overflow boundcpy WrongSizeParam\Path 19:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=34 |
| Status | New |

The size of the buffer used by parsing_error in Namespace1792866305, at line 1917 of freeswitch@@sofia-sip-v1.13.6-CVE-2022-31003-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parsing_error passes to Namespace1792866305, at line 1917 of freeswitch@@sofia-sip-v1.13.6-CVE-2022-31003-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.6-CVE-2022-31003-TP.c | freeswitch@@sofia-sip-v1.13.6-CVE-2022-31003-TP.c |
| Line | 1922 | 1922 |
| Object | Namespace1792866305 | Namespace1792866305 |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.6-CVE-2022-31003-TP.c
Method static int parsing_error(sdp_parser_t *p, char const *fmt, ...)

```
....
1922.      memset(p->pr_error, 0, sizeof(p->pr_error));
```

### Buffer Overflow boundcpy WrongSizeParam\Path 20:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=35 |
| Status | New |

The size of the buffer used by crypto_rsa_common in length, at line 96 of FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that crypto_rsa_common passes to length, at line 96 of FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c |
| Line | 116 | 116 |
| Object | length | length |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c
Method static int crypto_rsa_common(const BYTE* input, int length, UINT32 key_length, const BYTE* modulus,

```
....
116.         memcpy(input_reverse, input, length);
```

### Buffer Overflow boundcpy WrongSizeParam\Path 21:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=36 |
| Status | New |

The size of the buffer used by avc444_compress in codedSize, at line 292 of FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that avc444_compress passes to codedSize, at line 292 of FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c |
| Line | 371 | 371 |
| Object | codedSize | codedSize |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c |
| Method | INT32 avc444_compress(H264_CONTEXT* h264, const BYTE* pSrcData, DWORD SrcFormat, UINT32 nSrcStep, |

```
....
371.                memcpy(h264->lumaData, coded, codedSize);
```

### Buffer Overflow boundcpy WrongSizeParam\Path 22:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=37 |
| Status | New |

The size of the buffer used by redirection_copy_data in len, at line 101 of FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that redirection_copy_data passes to len, at line 101 of FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Line | 113 | 113 |
| Object | len | len |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Method | static BOOL redirection_copy_data(BYTE** dst, UINT32* plen, const BYTE* str, size_t len) |

```
....
113.        memcpy(*dst, str, len);
```

### Buffer Overflow boundcpy WrongSizeParam\Path 23:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=38](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=38) |
| Status | New |

The size of the buffer used by rdp_redirection_read_base64_wchar in bplen, at line 243 of FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that rdp_redirection_read_base64_wchar passes to bplen, at line 243 of FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Line | 279 | 279 |
| Object | bplen | bplen |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Method | static BOOL rdp_redirection_read_base64_wchar(UINT32 flag, wStream* s, UINT32* pLength, |

```
....
279.            memcpy(&(*pData)[wpos], bptr, bplen);
```

### Buffer Overflow boundcpy WrongSizeParam\Path 24:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=39](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=39) |
| Status | New |

The size of the buffer used by avc444_compress in codedSize, at line 296 of FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that avc444_compress passes to codedSize, at line 296 of FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c |
| Line | 375 | 375 |
| Object | codedSize | codedSize |

| | |
|---|---|
| Code Snippet | |
| File Name | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c |
| Method | INT32 avc444_compress(H264_CONTEXT* h264, const BYTE* pSrcData, DWORD SrcFormat, UINT32 nSrcStep, |

```
....
375.              memcpy(h264->lumaData, coded, codedSize);
```

### Buffer Overflow boundcpy WrongSizeParam\Path 25:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=40 |
| Status | New |

The size of the buffer used by redirection_copy_data in len, at line 103 of FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that redirection_copy_data passes to len, at line 103 of FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Line | 115 | 115 |
| Object | len | len |

| | |
|---|---|
| Code Snippet | |
| File Name | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Method | static BOOL redirection_copy_data(BYTE** dst, UINT32* plen, const BYTE* str, size_t len) |

```
....
115.         memcpy(*dst, str, len);
```

### Buffer Overflow boundcpy WrongSizeParam\Path 26:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=41 |
| Status | New |

The size of the buffer used by rdp_redirection_read_base64_wchar in bplen, at line 245 of FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that rdp_redirection_read_base64_wchar passes to bplen, at line 245 of FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |

| | | |
|---|---|---|
| Line | 281 | 281 |
| Object | bplen | bplen |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c
Method static BOOL rdp_redirection_read_base64_wchar(UINT32 flag, wStream* s, UINT32* pLength,

```
....
281.             memcpy(&(*pData)[wpos], bptr, bplen);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 27:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=42](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=42) |
| Status | New |

The size of the buffer used by avc444_compress in codedSize, at line 298 of FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that avc444_compress passes to codedSize, at line 298 of FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c |
| Line | 377 | 377 |
| Object | codedSize | codedSize |

Code Snippet
File Name FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c
Method INT32 avc444_compress(H264_CONTEXT* h264, const BYTE* pSrcData, DWORD SrcFormat, UINT32 nSrcStep,

```
....
377.             memcpy(h264->lumaData, coded, codedSize);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 28:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=43](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=43) |
| Status | New |

The size of the buffer used by redirection_copy_data in len, at line 103 of FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that redirection_copy_data passes to len, at line 103 of FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Line | 115 | 115 |
| Object | len | len |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Method | static BOOL redirection_copy_data(BYTE** dst, UINT32* plen, const BYTE* str, size_t len) |

```
....
115.        memcpy(*dst, str, len);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 29:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=44 |
| Status | New |

The size of the buffer used by rdp_redirection_read_base64_wchar in bplen, at line 234 of FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that rdp_redirection_read_base64_wchar passes to bplen, at line 234 of FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Line | 270 | 270 |
| Object | bplen | bplen |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Method | static BOOL rdp_redirection_read_base64_wchar(UINT32 flag, wStream* s, UINT32* pLength, |

```
....
270.            memcpy(&(*pData)[wpos], bptr, bplen);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 30:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=45 |
| Status | New |

The size of the buffer used by avc444_compress in codedSize, at line 336 of FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that avc444_compress passes to codedSize, at line 336 of FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c |
| Line | 415 | 415 |
| Object | codedSize | codedSize |

Code Snippet
File Name FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c
Method INT32 avc444_compress(H264_CONTEXT* h264, const BYTE* pSrcData, DWORD SrcFormat, UINT32 nSrcStep,

```
....
415.                memcpy(h264->lumaData, coded, codedSize);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 31:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=46 |
| Status | New |

The size of the buffer used by mbedtls_x509_set_extension in val_len, at line 213 of FreeRTOS@@FreeRTOS-Kernel-V10.3.0-CVE-2024-23775-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that mbedtls_x509_set_extension passes to val_len, at line 213 of FreeRTOS@@FreeRTOS-Kernel-V10.3.0-CVE-2024-23775-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.3.0-CVE-2024-23775-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.3.0-CVE-2024-23775-TP.c |
| Line | 225 | 225 |
| Object | val_len | val_len |

Code Snippet
File Name FreeRTOS@@FreeRTOS-Kernel-V10.3.0-CVE-2024-23775-TP.c
Method int mbedtls_x509_set_extension( mbedtls_asn1_named_data **head, const char *oid, size_t oid_len,

```
....
225.        memcpy( cur->val.p + 1, val, val_len );
```

**Buffer Overflow boundcpy WrongSizeParam\Path 32:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=47 |
| Status | New |

The size of the buffer used by mbedtls_x509_write_sig in len, at line 294 of FreeRTOS@@FreeRTOS-Kernel-V10.3.0-CVE-2024-23775-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that mbedtls_x509_write_sig passes to len, at line 294 of FreeRTOS@@FreeRTOS-Kernel-V10.3.0-CVE-2024-23775-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.3.0-CVE-2024-23775-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.3.0-CVE-2024-23775-TP.c |
| Line | 306 | 306 |
| Object | len | len |

Code Snippet

| | |
|---|---|
| File Name | FreeRTOS@@FreeRTOS-Kernel-V10.3.0-CVE-2024-23775-TP.c |
| Method | int mbedtls_x509_write_sig( unsigned char **p, unsigned char *start, |

```
....
306.        memcpy( *p, sig, len );
```

**Buffer Overflow boundcpy WrongSizeParam\Path 33:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=48 |
| Status | New |

The size of the buffer used by stun_parse_message in STUN_TID_BYTES, at line 82 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_message passes to STUN_TID_BYTES, at line 82 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 92 | 92 |
| Object | STUN_TID_BYTES | STUN_TID_BYTES |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Method | int stun_parse_message(stun_msg_t *msg) |

```
....
92.     memcpy(msg->stun_hdr.tran_id, p + 4, STUN_TID_BYTES);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 34:**

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

| | |
|---|---|
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=49](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=49) |
| Status | New |

The size of the buffer used by stun_parse_attribute in len, at line 114 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_attribute passes to len, at line 114 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 182 | 182 |
| Object | len | len |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Method | int stun_parse_attribute(stun_msg_t *msg, unsigned char *p) |

```
....
182.      memcpy(attr->enc_buf.data, p, len);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 35:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=50](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=50) |
| Status | New |

The size of the buffer used by stun_parse_attr_buffer in len, at line 270 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_attr_buffer passes to len, at line 270 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 276 | 276 |
| Object | len | len |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Method | int stun_parse_attr_buffer(stun_attr_t *attr, const unsigned char *p, unsigned len) |

```
....
276.    memcpy(buf->data, p, len);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 36:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=51 |
| Status | New |

The size of the buffer used by stun_copy_buffer in p, at line 314 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_copy_buffer passes to p, at line 314 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 318 | 318 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_copy_buffer(stun_buffer_t *p, stun_buffer_t *p2) {

```
....
318.     memcpy(p->data, p2->data, p->size);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 37:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=52 |
| Status | New |

The size of the buffer used by stun_encode_buffer in a, at line 420 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_encode_buffer passes to a, at line 420 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 429 | 429 |
| Object | a | a |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_encode_buffer(stun_attr_t *attr) {

```
....
429.     memcpy(attr->enc_buf.data+4, a->data, a->size);
```

### Buffer Overflow boundcpy WrongSizeParam\Path 38:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=53 |
| Status | New |

The size of the buffer used by stun_validate_message_integrity in len, at line 499 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_validate_message_integrity passes to len, at line 499 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 529 | 529 |
| Object | len | len |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_validate_message_integrity(stun_msg_t *msg, stun_buffer_t *pwd)

```
....
529.    memcpy(padded_text, msg->enc_buf.data, len);
```

### Buffer Overflow boundcpy WrongSizeParam\Path 39:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=54 |
| Status | New |

The size of the buffer used by stun_encode_message in STUN_TID_BYTES, at line 660 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_encode_message passes to STUN_TID_BYTES, at line 660 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 724 | 724 |
| Object | STUN_TID_BYTES | STUN_TID_BYTES |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_encode_message(stun_msg_t *msg, stun_buffer_t *pwd) {

```
....
724.        memcpy(buf + 4, msg->stun_hdr.tran_id, STUN_TID_BYTES);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 40:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=55](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=55) |
| Status | New |

The size of the buffer used by stun_encode_message in attr, at line 660 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_encode_message passes to attr, at line 660 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 733 | 733 |
| Object | attr | attr |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Method | int stun_encode_message(stun_msg_t *msg, stun_buffer_t *pwd) { |

```
....
733.          memcpy(buf+len, (void *)attr->enc_buf.data, attr-
>enc_buf.size);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 41:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=56](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=56) |
| Status | New |

The size of the buffer used by stun_encode_message in msg_int, at line 660 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_encode_message passes to msg_int, at line 660 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 746 | 746 |
| Object | msg_int | msg_int |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Method | int stun_encode_message(stun_msg_t *msg, stun_buffer_t *pwd) { |

```
....
746.            msg_int->enc_buf.size);
```

### Buffer Overflow boundcpy WrongSizeParam\Path 42:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=57 |
| Status | New |

The size of the buffer used by stun_parse_message in STUN_TID_BYTES, at line 82 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_message passes to STUN_TID_BYTES, at line 82 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 92 | 92 |
| Object | STUN_TID_BYTES | STUN_TID_BYTES |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Method | int stun_parse_message(stun_msg_t *msg) |

```
....
92.    memcpy(msg->stun_hdr.tran_id, p + 4, STUN_TID_BYTES);
```

### Buffer Overflow boundcpy WrongSizeParam\Path 43:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=58 |
| Status | New |

The size of the buffer used by stun_parse_attribute in len, at line 114 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_attribute passes to len, at line 114 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 182 | 182 |
| Object | len | len |

| | |
|---|---|
| Code Snippet | |
| File Name | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Method | int stun_parse_attribute(stun_msg_t *msg, unsigned char *p) |

```
....
182.        memcpy(attr->enc_buf.data, p, len);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 44:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=59 |
| Status | New |

The size of the buffer used by stun_parse_attr_buffer in len, at line 270 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_parse_attr_buffer passes to len, at line 270 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 276 | 276 |
| Object | len | len |

| | |
|---|---|
| Code Snippet | |
| File Name | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Method | int stun_parse_attr_buffer(stun_attr_t *attr, const unsigned char *p, unsigned len) |

```
....
276.      memcpy(buf->data, p, len);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 45:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=60 |
| Status | New |

The size of the buffer used by stun_copy_buffer in p, at line 314 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_copy_buffer passes to p, at line 314 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 318 | 318 |

| | | |
|---|---|---|
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_copy_buffer(stun_buffer_t *p, stun_buffer_t *p2) {

```
....
318.    memcpy(p->data, p2->data, p->size);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 46:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=61 |
| Status | New |

The size of the buffer used by stun_encode_buffer in a, at line 420 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_encode_buffer passes to a, at line 420 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 429 | 429 |
| Object | a | a |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_encode_buffer(stun_attr_t *attr) {

```
....
429.    memcpy(attr->enc_buf.data+4, a->data, a->size);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 47:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=62 |
| Status | New |

The size of the buffer used by stun_validate_message_integrity in len, at line 499 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_validate_message_integrity passes to len, at line 499 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |

| | | |
|---|---|---|
| Line | 529 | 529 |
| Object | len | len |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_validate_message_integrity(stun_msg_t *msg, stun_buffer_t *pwd)

```
....
529.    memcpy(padded_text, msg->enc_buf.data, len);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 48:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=63](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=63) |
| Status | New |

The size of the buffer used by stun_encode_message in STUN_TID_BYTES, at line 660 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_encode_message passes to STUN_TID_BYTES, at line 660 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 724 | 724 |
| Object | STUN_TID_BYTES | STUN_TID_BYTES |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_encode_message(stun_msg_t *msg, stun_buffer_t *pwd) {

```
....
724.      memcpy(buf + 4, msg->stun_hdr.tran_id, STUN_TID_BYTES);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 49:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=64](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=64) |
| Status | New |

The size of the buffer used by stun_encode_message in attr, at line 660 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_encode_message passes to attr, at line 660 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, to overwrite the target buffer.

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

| | | |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 733 | 733 |
| Object | attr | attr |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_encode_message(stun_msg_t *msg, stun_buffer_t *pwd) {

```
....
733.        memcpy(buf+len, (void *)attr->enc_buf.data, attr-
>enc_buf.size);
```

**Buffer Overflow boundcpy WrongSizeParam\Path 50:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=65 |
| Status | New |

The size of the buffer used by stun_encode_message in msg_int, at line 660 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stun_encode_message passes to msg_int, at line 660 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c, to overwrite the target buffer.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 746 | 746 |
| Object | msg_int | msg_int |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_encode_message(stun_msg_t *msg, stun_buffer_t *pwd) {

```
....
746.             msg_int->enc_buf.size);
```

## 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=1000019&projectid=14&pathid=303 |
| Status | New |

The function size in FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c at line 2040 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 | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 2134 | 2134 |
| Object | size | size |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2134.                 groupDsc = (FILEGROUPDESCRIPTORW*)malloc(size);
```

**Wrong Size t Allocation\Path 2:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=304 |
| Status | New |

The function size in FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c at line 2040 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 | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 2166 | 2166 |
| Object | size | size |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2166.                       buff = malloc(size);
```

**Wrong Size t Allocation\Path 3:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=305 |

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

The function size in FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c at line 2041 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 | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 2135 | 2135 |
| Object | size | size |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2135.           groupDsc = (FILEGROUPDESCRIPTORW*)malloc(size);
```

**Wrong Size t Allocation\Path 4:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=306 |
| Status | New |

The function size in FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c at line 2041 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 | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 2167 | 2167 |
| Object | size | size |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2167.               buff = malloc(size);
```

**Wrong Size t Allocation\Path 5:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14 |

| | |
|---|---|
| | &pathid=307 |
| Status | New |

The function size in FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c at line 2041 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 | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 2135 | 2135 |
| Object | size | size |

Code Snippet
File Name FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2135.           groupDsc = (FILEGROUPDESCRIPTORW*)malloc(size);
```

**Wrong Size t Allocation\Path 6:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=308 |
| Status | New |

The function size in FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c at line 2041 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 | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 2167 | 2167 |
| Object | size | size |

Code Snippet
File Name FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2167.                   buff = malloc(size);
```

**Wrong Size t Allocation\Path 7:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=309 |
| Status | New |

The function len in FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c at line 101 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 | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Line | 110 | 110 |
| Object | len | len |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Method | static BOOL redirection_copy_data(BYTE** dst, UINT32* plen, const BYTE* str, size_t len) |

```
....
110.        *dst = malloc(len);
```

**Wrong Size t Allocation\Path 8:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=310 |
| Status | New |

The function len in FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c at line 103 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Line | 112 | 112 |
| Object | len | len |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Method | static BOOL redirection_copy_data(BYTE** dst, UINT32* plen, const BYTE* str, size_t len) |

```
....
112.        *dst = malloc(len);
```

**Wrong Size t Allocation\Path 9:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=311 |
| Status | New |

The function len in FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c at line 103 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Line | 112 | 112 |
| Object | len | len |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Method | static BOOL redirection_copy_data(BYTE** dst, UINT32* plen, const BYTE* str, size_t len) |

```
....
112.        *dst = malloc(len);
```

**Wrong Size t Allocation\Path 10:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=312 |
| Status | New |

The function nSize in FreeRDP@@FreeRDP-2.0.0-CVE-2020-11097-TP.c at line 228 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 | FreeRDP@@FreeRDP-2.0.0-CVE-2020-11097-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-11097-TP.c |
| Line | 238 | 238 |
| Object | nSize | nSize |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-11097-TP.c |
| Method | static int ntlm_get_target_computer_name(PUNICODE_STRING pName, COMPUTER_NAME_FORMAT type) |

```
....
238.        computerName = calloc(nSize, sizeof(CHAR));
```

**Wrong Size t Allocation\Path 11:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=313 |
| Status | New |

The function maximum_size in FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c at line 70 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 | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 80 | 80 |
| Object | maximum_size | maximum_size |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c
Method static char* rdp_info_package_flags_description(UINT32 flags)

```
....
80.    result = calloc(maximum_size, sizeof(char));
```

**Wrong Size t Allocation\Path 12:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=314 |
| Status | New |

The function nullSize in FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c at line 70 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 | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 106 | 106 |
| Object | nullSize | nullSize |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c

| | |
|---|---|
| Method | static BOOL rdp_read_info_null_string(UINT32 flags, wStream* s, size_t cbLen, CHAR** dst, |

```
....
106.                    ret = calloc(cbLen + 1, nullSize);
```

### Wrong Size t Allocation\Path 13:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=315](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=315) |
| Status | New |

The function maximum_size in FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c at line 118 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 | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 128 | 128 |
| Object | maximum_size | maximum_size |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Method | static char* rdp_info_package_flags_description(UINT32 flags) |

```
....
128.        result = calloc(maximum_size, sizeof(char));
```

### Wrong Size t Allocation\Path 14:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=316](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=316) |
| Status | New |

The function nullSize in FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c at line 466 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 | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 507 | 507 |
| Object | nullSize | nullSize |

|  |  |
|---|---|
| Code Snippet |  |
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_read_info_string(UINT32 flags, wStream* s, size_t cbLenNonNull, CHAR** dst, |

```
....
507.                    ret = calloc(cbLenNonNull + 1, nullSize);
```

### Wrong Size t Allocation\Path 15:

|  |  |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=317 |
| Status | New |

The function nullSize in FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp at line 70 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 | FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp | FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp |
| Line | 106 | 106 |
| Object | nullSize | nullSize |

|  |  |
|---|---|
| Code Snippet |  |
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp |
| Method | static BOOL rdp_read_info_null_string(UINT32 flags, wStream* s, size_t cbLen, CHAR** dst, |

```
....
106.                    ret = calloc(cbLen + 1, nullSize);
```

### Wrong Size t Allocation\Path 16:

|  |  |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=318 |
| Status | New |

The function maximum_size in FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp at line 118 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 | FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp | FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp |
| Line | 128 | 128 |

| | | |
|---|---|---|
| Object | maximum_size | maximum_size |

Code Snippet
File Name FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp
Method static char* rdp_info_package_flags_description(UINT32 flags)

```
....
128.         result = calloc(maximum_size, sizeof(char));
```

**Wrong Size t Allocation\Path 17:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=319 |
| Status | New |

The function nullSize in FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp at line 466 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 | FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp | FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp |
| Line | 507 | 507 |
| Object | nullSize | nullSize |

Code Snippet
File Name FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp
Method static BOOL rdp_read_info_string(UINT32 flags, wStream* s, size_t cbLenNonNull, CHAR** dst,

```
....
507.                    ret = calloc(cbLenNonNull + 1, nullSize);
```

**Wrong Size t Allocation\Path 18:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=320 |
| Status | New |

The function nullSize in FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c at line 70 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 | FreeRDP@@FreeRDP-2.4.0-CVE-2024- | FreeRDP@@FreeRDP-2.4.0-CVE-2024- |

| | | |
|---|---|---|
| | 32661-TP.c | 32661-TP.c |
| Line | 106 | 106 |
| Object | nullSize | nullSize |

Code Snippet
File Name FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c
Method static BOOL rdp_read_info_null_string(UINT32 flags, wStream* s, size_t cbLen, CHAR** dst,

```
....
106.                        ret = calloc(cbLen + 1, nullSize);
```

**Wrong Size t Allocation\Path 19:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=321 |
| Status | New |

The function maximum_size in FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c at line 118 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 | FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c |
| Line | 128 | 128 |
| Object | maximum_size | maximum_size |

Code Snippet
File Name FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c
Method static char* rdp_info_package_flags_description(UINT32 flags)

```
....
128.         result = calloc(maximum_size, sizeof(char));
```

**Wrong Size t Allocation\Path 20:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=322 |
| Status | New |

The function nullSize in FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c at line 466 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 | FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c |
| Line | 507 | 507 |
| Object | nullSize | nullSize |

Code Snippet
File Name FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c
Method static BOOL rdp_read_info_string(UINT32 flags, wStream* s, size_t cbLenNonNull, CHAR** dst,

```
....
507.                    ret = calloc(cbLenNonNull + 1, nullSize);
```

**Wrong Size t Allocation\Path 21:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=323 |
| Status | New |

The function nullSize in FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c at line 70 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 | FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c |
| Line | 106 | 106 |
| Object | nullSize | nullSize |

Code Snippet
File Name FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c
Method static BOOL rdp_read_info_null_string(UINT32 flags, wStream* s, size_t cbLen, CHAR** dst,

```
....
106.                    ret = calloc(cbLen + 1, nullSize);
```

**Wrong Size t Allocation\Path 22:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=324 |
| Status | New |

The function maximum_size in FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c at line 118 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 | FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c |
| Line | 128 | 128 |
| Object | maximum_size | maximum_size |

Code Snippet
File Name FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c
Method static char* rdp_info_package_flags_description(UINT32 flags)

```
....
128.        result = calloc(maximum_size, sizeof(char));
```

### Wrong Size t Allocation\Path 23:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=325 |
| Status | New |

The function nullSize in FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c at line 466 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 | FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c |
| Line | 507 | 507 |
| Object | nullSize | nullSize |

Code Snippet
File Name FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c
Method static BOOL rdp_read_info_string(UINT32 flags, wStream* s, size_t cbLenNonNull, CHAR** dst,

```
....
507.                ret = calloc(cbLenNonNull + 1, nullSize);
```

### Wrong Size t Allocation\Path 24:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=326 |

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

The function nullSize in FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c at line 70 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 | FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c |
| Line | 106 | 106 |
| Object | nullSize | nullSize |

Code Snippet

File Name FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c

Method static BOOL rdp_read_info_null_string(UINT32 flags, wStream* s, size_t cbLen, CHAR** dst,

```
....
106.                    ret = calloc(cbLen + 1, nullSize);
```

**Wrong Size t Allocation\Path 25:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=327 |
| Status | New |

The function maximum_size in FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c at line 118 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 | FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c |
| Line | 128 | 128 |
| Object | maximum_size | maximum_size |

Code Snippet

File Name FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c

Method static char* rdp_info_package_flags_description(UINT32 flags)

```
....
128.          result = calloc(maximum_size, sizeof(char));
```

**Wrong Size t Allocation\Path 26:**

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

| | |
|---|---|
| | [PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=328](PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=328) |
| Status | New |

The function nullSize in FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c at line 466 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 | FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c |
| Line | 507 | 507 |
| Object | nullSize | nullSize |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_read_info_string(UINT32 flags, wStream* s, size_t cbLenNonNull, CHAR** dst, |

```
....
507.                    ret = calloc(cbLenNonNull + 1, nullSize);
```

**Wrong Size t Allocation\Path 27:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=329](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=329) |
| Status | New |

The function nullSize in FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c at line 72 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 | FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c |
| Line | 108 | 108 |
| Object | nullSize | nullSize |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_read_info_null_string(UINT32 flags, wStream* s, size_t cbLen, CHAR** dst, |

```
....
108.                    ret = calloc(cbLen + 1, nullSize);
```

**Wrong Size t Allocation\Path 28:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=330 |
| Status | New |

The function nullSize in FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c at line 459 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 | FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c |
| Line | 500 | 500 |
| Object | nullSize | nullSize |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_read_info_string(UINT32 flags, wStream* s, size_t cbLenNonNull, CHAR** dst, |

```
....
500.                    ret = calloc(cbLenNonNull + 1, nullSize);
```

**Wrong Size t Allocation\Path 29:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=331 |
| Status | New |

The function nullSize in FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c at line 72 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 | FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c |
| Line | 108 | 108 |
| Object | nullSize | nullSize |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_read_info_null_string(UINT32 flags, wStream* s, size_t cbLen, CHAR** dst, |

```
....
108.                ret = calloc(cbLen + 1, nullSize);
```

**Wrong Size t Allocation\Path 30:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=332 |
| Status | New |

The function nullSize in FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c at line 459 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 | FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c |
| Line | 500 | 500 |
| Object | nullSize | nullSize |

Code Snippet
File Name FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c
Method static BOOL rdp_read_info_string(UINT32 flags, wStream* s, size_t cbLenNonNull, CHAR** dst,

```
....
500.                ret = calloc(cbLenNonNull + 1, nullSize);
```

**Wrong Size t Allocation\Path 31:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=333 |
| Status | New |

The function len in FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c at line 118 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 | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Line | 125 | 125 |
| Object | len | len |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Method | static BOOL redirection_copy_array(char*** dst, UINT32* plen, const char** str, size_t len) |

```
....
125.          *dst = calloc(len, sizeof(char));
```

**Wrong Size t Allocation\Path 32:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=334](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=334) |
| Status | New |

The function utf8_len in FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c at line 243 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 | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Line | 262 | 262 |
| Object | utf8_len | utf8_len |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Method | static BOOL rdp_redirection_read_base64_wchar(UINT32 flag, wStream* s, UINT32* pLength, |

```
....
262.          *pData = calloc(utf8_len, sizeof(BYTE));
```

**Wrong Size t Allocation\Path 33:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=335](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=335) |
| Status | New |

The function len in FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c at line 120 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Line | 127 | 127 |

| | | |
|---|---|---|
| Object | len | len |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c
Method static BOOL redirection_copy_array(char*** dst, UINT32* plen, const char** str, size_t len)

```
....
127.          *dst = calloc(len, sizeof(char));
```

**Wrong Size t Allocation\Path 34:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=336 |
| Status | New |

The function utf8_len in FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c at line 245 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 | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Line | 264 | 264 |
| Object | utf8_len | utf8_len |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c
Method static BOOL rdp_redirection_read_base64_wchar(UINT32 flag, wStream* s, UINT32* pLength,

```
....
264.          *pData = calloc(utf8_len, sizeof(BYTE));
```

**Wrong Size t Allocation\Path 35:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=337 |
| Status | New |

The function len in FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c at line 120 assigns an incorrectly calculated size to a buffer, resulting in a mismatch between the value being written and the size of the buffer it is being written into.

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

| | | |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Line | 127 | 127 |
| Object | len | len |

Code Snippet
File Name FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c
Method static BOOL redirection_copy_array(char*** dst, UINT32* plen, const char** str, size_t len)

```
....
127.        *dst = calloc(len, sizeof(char*));
```

**Wrong Size t Allocation\Path 36:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=338 |
| Status | New |

The function utf8_len in FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c at line 234 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 | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Line | 253 | 253 |
| Object | utf8_len | utf8_len |

Code Snippet
File Name FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c
Method static BOOL rdp_redirection_read_base64_wchar(UINT32 flag, wStream* s, UINT32* pLength,

```
....
253.        *pData = calloc(utf8_len, sizeof(BYTE));
```

**Wrong Size t Allocation\Path 37:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=339 |
| Status | New |

The function address_size in freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c at line 763 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 | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 786 | 786 |
| Object | address_size | address_size |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method char *stun_determine_ip_address(int family)

```
....
786.    local_ip_address = malloc(address_size + 1);
```

**Wrong Size t Allocation\Path 38:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=340 |
| Status | New |

The function address_size in freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c at line 763 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 | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 786 | 786 |
| Object | address_size | address_size |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method char *stun_determine_ip_address(int family)

```
....
786.    local_ip_address = malloc(address_size + 1);
```

**Wrong Size t Allocation\Path 39:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=341 |
| Status | New |

The function address_size in freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c at line 763 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 | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 786 | 786 |
| Object | address_size | address_size |

Code Snippet
File Name	freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c
Method	char *stun_determine_ip_address(int family)

```
....
786.      local_ip_address = malloc(address_size + 1);
```

### Wrong Size t Allocation\Path 40:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=342 |
| Status | New |

The function address_size in freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c at line 763 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 | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c |
| Line | 786 | 786 |
| Object | address_size | address_size |

Code Snippet
File Name	freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c
Method	char *stun_determine_ip_address(int family)

```
....
786.      local_ip_address = malloc(address_size + 1);
```

### Wrong Size t Allocation\Path 41:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=343 |

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

The function address_size in freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c at line 763 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 | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c |
| Line | 786 | 786 |
| Object | address_size | address_size |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c
Method char *stun_determine_ip_address(int family)

```
....
786.     local_ip_address = malloc(address_size + 1);
```

**Wrong Size t Allocation\Path 42:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=344 |
| Status | New |

The function new_size in FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c at line 1136 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 | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1147 | 1147 |
| Object | new_size | new_size |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method static void map_ensure_capacity(wfClipboard* clipboard)

```
....
1147.                (formatMapping*)realloc(clipboard-
>format_mappings, sizeof(formatMapping) * new_size);
```

**Wrong Size t Allocation\Path 43:**

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=345 |
| Status | New |

The function new_size in FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c at line 1696 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 | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1708 | 1708 |
| Object | new_size | new_size |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Method | static BOOL wf_cliprdr_array_ensure_capacity(wfClipboard* clipboard) |

```
....
1708.                                                       new_size *
sizeof(FILEDESCRIPTORW*));
```

**Wrong Size t Allocation\Path 44:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=346 |
| Status | New |

The function new_size in FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c at line 1696 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 | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1713 | 1713 |
| Object | new_size | new_size |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Method | static BOOL wf_cliprdr_array_ensure_capacity(wfClipboard* clipboard) |

```
....
1713.               new_name = (WCHAR**)realloc(clipboard->file_names,
new_size * sizeof(WCHAR*));
```

**Wrong Size t Allocation\Path 45:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=347 |
| Status | New |

The function new_size in FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c at line 1136 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 | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1147 | 1147 |
| Object | new_size | new_size |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c
Method static void map_ensure_capacity(wfClipboard* clipboard)

```
....
1147.                  (formatMapping*)realloc(clipboard-
>format_mappings, sizeof(formatMapping) * new_size);
```

**Wrong Size t Allocation\Path 46:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=348 |
| Status | New |

The function new_size in FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c at line 1697 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 | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1709 | 1709 |
| Object | new_size | new_size |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c
Method static BOOL wf_cliprdr_array_ensure_capacity(wfClipboard* clipboard)

```
....
1709.                                                     new_size *
sizeof(FILEDESCRIPTORW*));
```

**Wrong Size t Allocation\Path 47:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=349](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=349) |
| Status | New |

The function new_size in FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c at line 1697 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 | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1714 | 1714 |
| Object | new_size | new_size |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c
Method static BOOL wf_cliprdr_array_ensure_capacity(wfClipboard* clipboard)

```
....
1714.                new_name = (WCHAR**)realloc(clipboard->file_names,
new_size * sizeof(WCHAR*));
```

**Wrong Size t Allocation\Path 48:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=350](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=350) |
| Status | New |

The function new_size in FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c at line 1136 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 | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 1147 | 1147 |
| Object | new_size | new_size |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Method | static void map_ensure_capacity(wfClipboard* clipboard) |

```
....
1147.                 (formatMapping*)realloc(clipboard-
>format_mappings, sizeof(formatMapping) * new_size);
```

### Wrong Size t Allocation\Path 49:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=351 |
| Status | New |

The function new_size in FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c at line 1697 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 | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 1709 | 1709 |
| Object | new_size | new_size |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Method | static BOOL wf_cliprdr_array_ensure_capacity(wfClipboard* clipboard) |

```
....
1709.                                                    new_size *
sizeof(FILEDESCRIPTORW*));
```

### Wrong Size t Allocation\Path 50:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=352 |
| Status | New |

The function new_size in FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c at line 1697 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 | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 1714 | 1714 |

| | | |
|---|---|---|
| Object | new_size | new_size |

Code Snippet
File Name FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c
Method static BOOL wf_cliprdr_array_ensure_capacity(wfClipboard* clipboard)

```
....
1714.              new_name = (WCHAR**)realloc(clipboard->file_names,
new_size * sizeof(WCHAR*));
```

## Use of a One Way Hash without a Salt

Query Path:
CPP\Cx\CPP Medium Threat\Use of a One Way Hash without a Salt Version:1

### Categories

FISMA 2014: Media Protection
NIST SP 800-53: SC-13 Cryptographic Protection (P1)

### Description

**Use of a One Way Hash without a Salt\Path 1:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1046 |
| Status | New |

The application protects passwords with HMAC in stun_encode_message_integrity, of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c at line 434, using a cryptographic hash padded_text. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 455 | 455 |
| Object | padded_text | HMAC |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_encode_message_integrity(stun_attr_t *attr,

```
....
455.       sha1_hmac = HMAC(EVP_sha1(), pwd->data, pwd->size,
padded_text, padded_len, NULL, &dig_len);
```

**Use of a One Way Hash without a Salt\Path 2:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1047 |

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

The application protects passwords with HMAC in stun_encode_message_integrity, of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c at line 434, using a cryptographic hash buf. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 733 | 455 |
| Object | buf | HMAC |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_encode_message(stun_msg_t *msg, stun_buffer_t *pwd) {

```
....
733.         memcpy(buf+len, (void *)attr->enc_buf.data, attr-
>enc_buf.size);
```

File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_encode_message_integrity(stun_attr_t *attr,

```
....
455.       sha1_hmac = HMAC(EVP_sha1(), pwd->data, pwd->size,
padded_text, padded_len, NULL, &dig_len);
```

### Use of a One Way Hash without a Salt\Path 3:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1048](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1048) |
| Status | New |

The application protects passwords with HMAC in stun_encode_message_integrity, of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c at line 434, using a cryptographic hash buf. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 458 | 458 |
| Object | buf | HMAC |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_encode_message_integrity(stun_attr_t *attr,

```
....
458.        sha1_hmac = HMAC(EVP_sha1(), pwd->data, pwd->size, buf, len,
NULL, &dig_len);
```

### Use of a One Way Hash without a Salt\Path 4:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1049 |
| Status | New |

The application protects passwords with HMAC in stun_validate_message_integrity, of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c at line 499, using a cryptographic hash padded_text. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 531 | 531 |
| Object | padded_text | HMAC |

Code Snippet
File Name    freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method    int stun_validate_message_integrity(stun_msg_t *msg, stun_buffer_t *pwd)

```
....
531.    memcpy(dig, HMAC(EVP_sha1(), pwd->data, pwd->size, padded_text,
padded_len, NULL, &dig_len), 20);
```

### Use of a One Way Hash without a Salt\Path 5:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1050 |
| Status | New |

The application protects passwords with HMAC in stun_encode_message_integrity, of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c at line 434, using a cryptographic hash padded_text. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 455 | 455 |
| Object | padded_text | HMAC |

Code Snippet

File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c

Method int stun_encode_message_integrity(stun_attr_t *attr,

```
....
455.       sha1_hmac = HMAC(EVP_sha1(), pwd->data, pwd->size,
padded_text, padded_len, NULL, &dig_len);
```

### Use of a One Way Hash without a Salt\Path 6:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1051 |
| Status | New |

The application protects passwords with HMAC in stun_encode_message_integrity, of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c at line 434, using a cryptographic hash buf. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 733 | 455 |
| Object | buf | HMAC |

Code Snippet

File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c

Method int stun_encode_message(stun_msg_t *msg, stun_buffer_t *pwd) {

```
....
733.         memcpy(buf+len, (void *)attr->enc_buf.data, attr-
>enc_buf.size);
```

File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c

Method int stun_encode_message_integrity(stun_attr_t *attr,

```
....
455.       sha1_hmac = HMAC(EVP_sha1(), pwd->data, pwd->size,
padded_text, padded_len, NULL, &dig_len);
```

### Use of a One Way Hash without a Salt\Path 7:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1052 |
| Status | New |

The application protects passwords with HMAC in stun_encode_message_integrity, of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c at line 434, using a cryptographic hash buf. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 458 | 458 |
| Object | buf | HMAC |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_encode_message_integrity(stun_attr_t *attr,

```
....
458.       sha1_hmac = HMAC(EVP_sha1(), pwd->data, pwd->size, buf, len,
NULL, &dig_len);
```

**Use of a One Way Hash without a Salt\Path 8:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1053 |
| Status | New |

The application protects passwords with HMAC in stun_validate_message_integrity, of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c at line 499, using a cryptographic hash padded_text. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 531 | 531 |
| Object | padded_text | HMAC |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_validate_message_integrity(stun_msg_t *msg, stun_buffer_t *pwd)

```
....
531.    memcpy(dig, HMAC(EVP_sha1(), pwd->data, pwd->size, padded_text,
padded_len, NULL, &dig_len), 20);
```

**Use of a One Way Hash without a Salt\Path 9:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14 |

| | |
|---|---|
| | &pathid=1054 |
| Status | New |

The application protects passwords with HMAC in stun_encode_message_integrity, of freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c at line 434, using a cryptographic hash padded_text. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 455 | 455 |
| Object | padded_text | HMAC |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c
Method int stun_encode_message_integrity(stun_attr_t *attr,

```
....
455.       sha1_hmac = HMAC(EVP_sha1(), pwd->data, pwd->size,
padded_text, padded_len, NULL, &dig_len);
```

### Use of a One Way Hash without a Salt\Path 10:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1055 |
| Status | New |

The application protects passwords with HMAC in stun_encode_message_integrity, of freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c at line 434, using a cryptographic hash buf. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 733 | 455 |
| Object | buf | HMAC |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c
Method int stun_encode_message(stun_msg_t *msg, stun_buffer_t *pwd) {

```
....
733.         memcpy(buf+len, (void *)attr->enc_buf.data, attr-
>enc_buf.size);
```

File Name freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c

|  |  |
|---|---|
| Method | int stun_encode_message_integrity(stun_attr_t *attr, |

```
....
455.        sha1_hmac = HMAC(EVP_sha1(), pwd->data, pwd->size,
padded_text, padded_len, NULL, &dig_len);
```

### Use of a One Way Hash without a Salt\Path 11:

|  |  |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1056](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1056) |
| Status | New |

The application protects passwords with HMAC in stun_encode_message_integrity, of freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c at line 434, using a cryptographic hash buf. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

|  | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 458 | 458 |
| Object | buf | HMAC |

Code Snippet

|  |  |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Method | int stun_encode_message_integrity(stun_attr_t *attr, |

```
....
458.        sha1_hmac = HMAC(EVP_sha1(), pwd->data, pwd->size, buf, len,
NULL, &dig_len);
```

### Use of a One Way Hash without a Salt\Path 12:

|  |  |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1057](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1057) |
| Status | New |

The application protects passwords with HMAC in stun_validate_message_integrity, of freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c at line 499, using a cryptographic hash padded_text. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

|  | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 531 | 531 |
| Object | padded_text | HMAC |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Method | int stun_validate_message_integrity(stun_msg_t *msg, stun_buffer_t *pwd) |

```
....
531.     memcpy(dig, HMAC(EVP_sha1(), pwd->data, pwd->size, padded_text,
padded_len, NULL, &dig_len), 20);
```

### Use of a One Way Hash without a Salt\Path 13:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1058 |
| Status | New |

The application protects passwords with HMAC in stun_encode_message_integrity, of freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c at line 434, using a cryptographic hash padded_text. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c |
| Line | 455 | 455 |
| Object | padded_text | HMAC |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c |
| Method | int stun_encode_message_integrity(stun_attr_t *attr, |

```
....
455.       sha1_hmac = HMAC(EVP_sha1(), pwd->data, pwd->size,
padded_text, padded_len, NULL, &dig_len);
```

### Use of a One Way Hash without a Salt\Path 14:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1059 |
| Status | New |

The application protects passwords with HMAC in stun_encode_message_integrity, of freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c at line 434, using a cryptographic hash buf. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c |

| | | |
|---|---|---|
| Line | 733 | 455 |
| Object | buf | HMAC |

Code Snippet

File Name freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c

Method int stun_encode_message(stun_msg_t *msg, stun_buffer_t *pwd) {

```
....
733.        memcpy(buf+len, (void *)attr->enc_buf.data, attr-
>enc_buf.size);
```

File Name freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c

Method int stun_encode_message_integrity(stun_attr_t *attr,

```
....
455.       sha1_hmac = HMAC(EVP_sha1(), pwd->data, pwd->size,
padded_text, padded_len, NULL, &dig_len);
```

### Use of a One Way Hash without a Salt\Path 15:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1060 |
| Status | New |

The application protects passwords with HMAC in stun_encode_message_integrity, of freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c at line 434, using a cryptographic hash buf. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c |
| Line | 458 | 458 |
| Object | buf | HMAC |

Code Snippet

File Name freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c

Method int stun_encode_message_integrity(stun_attr_t *attr,

```
....
458.       sha1_hmac = HMAC(EVP_sha1(), pwd->data, pwd->size, buf, len,
NULL, &dig_len);
```

### Use of a One Way Hash without a Salt\Path 16:

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1061 |
| Status | New |

The application protects passwords with HMAC in stun_validate_message_integrity, of freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c at line 499, using a cryptographic hash padded_text. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c |
| Line | 531 | 531 |
| Object | padded_text | HMAC |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c
Method int stun_validate_message_integrity(stun_msg_t *msg, stun_buffer_t *pwd)

```
....
531.     memcpy(dig, HMAC(EVP_sha1(), pwd->data, pwd->size, padded_text,
padded_len, NULL, &dig_len), 20);
```

### Use of a One Way Hash without a Salt\Path 17:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1062 |
| Status | New |

The application protects passwords with HMAC in stun_encode_message_integrity, of freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c at line 434, using a cryptographic hash padded_text. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c |
| Line | 455 | 455 |
| Object | padded_text | HMAC |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c
Method int stun_encode_message_integrity(stun_attr_t *attr,

```
....
455.       sha1_hmac = HMAC(EVP_sha1(), pwd->data, pwd->size,
padded_text, padded_len, NULL, &dig_len);
```

### Use of a One Way Hash without a Salt\Path 18:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1063 |
| Status | New |

The application protects passwords with HMAC in stun_encode_message_integrity, of freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c at line 434, using a cryptographic hash buf. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c |
| Line | 733 | 455 |
| Object | buf | HMAC |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c
Method int stun_encode_message(stun_msg_t *msg, stun_buffer_t *pwd) {

```
....
733.         memcpy(buf+len, (void *)attr->enc_buf.data, attr->enc_buf.size);
```

File Name freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c
Method int stun_encode_message_integrity(stun_attr_t *attr,

```
....
455.       sha1_hmac = HMAC(EVP_sha1(), pwd->data, pwd->size, padded_text, padded_len, NULL, &dig_len);
```

### Use of a One Way Hash without a Salt\Path 19:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1064 |
| Status | New |

The application protects passwords with HMAC in stun_encode_message_integrity, of freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c at line 434, using a cryptographic hash buf. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c |
| Line | 458 | 458 |

| | | |
|---|---|---|
| Object | buf | HMAC |

Code Snippet
File Name	freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c
Method	int stun_encode_message_integrity(stun_attr_t *attr,

```
....
458.        sha1_hmac = HMAC(EVP_sha1(), pwd->data, pwd->size, buf, len,
NULL, &dig_len);
```

**Use of a One Way Hash without a Salt\Path 20:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1065 |
| Status | New |

The application protects passwords with HMAC in stun_validate_message_integrity, of freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c at line 499, using a cryptographic hash padded_text. However, the code does not salt the hash with an unpredictable, random value, allowing an attacker to reverse the hash value.

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c |
| Line | 531 | 531 |
| Object | padded_text | HMAC |

Code Snippet
File Name	freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c
Method	int stun_validate_message_integrity(stun_msg_t *msg, stun_buffer_t *pwd)

```
....
531.     memcpy(dig, HMAC(EVP_sha1(), pwd->data, pwd->size, padded_text,
padded_len, NULL, &dig_len), 20);
```

## 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=1000019&projectid=14&pathid=900 |
| Status | New |

Method rdp_write_info_packet at line 721 of FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c defines cbPassword, which is designated to contain user passwords. However, while plaintext passwords are later assigned to cbPassword, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 730 | 730 |
| Object | cbPassword | cbPassword |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
730.        UINT16 cbPassword = 0;
```

**Heap Inspection\Path 2:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=901 |
| Status | New |

Method rdp_write_info_packet at line 609 of FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c defines cbPassword, which is designated to contain user passwords. However, while plaintext passwords are later assigned to cbPassword, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 618 | 618 |
| Object | cbPassword | cbPassword |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
618.        UINT16 cbPassword = 0;
```

**Heap Inspection\Path 3:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14 |

| | |
|---|---|
| | &pathid=902 |
| Status | New |

Method rdp_write_info_packet at line 609 of FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp defines cbPassword, which is designated to contain user passwords. However, while plaintext passwords are later assigned to cbPassword, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp | FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp |
| Line | 618 | 618 |
| Object | cbPassword | cbPassword |

Code Snippet
File Name FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
618.        UINT16 cbPassword = 0;
```

**Heap Inspection\Path 4:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=903 |
| Status | New |

Method rdp_write_info_packet at line 609 of FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c defines cbPassword, which is designated to contain user passwords. However, while plaintext passwords are later assigned to cbPassword, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c |
| Line | 618 | 618 |
| Object | cbPassword | cbPassword |

Code Snippet
File Name FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
618.        UINT16 cbPassword = 0;
```

**Heap Inspection\Path 5:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=904 |

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

Method rdp_write_info_packet at line 609 of FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c defines cbPassword, which is designated to contain user passwords. However, while plaintext passwords are later assigned to cbPassword, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c |
| Line | 618 | 618 |
| Object | cbPassword | cbPassword |

Code Snippet
File Name FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
618.        UINT16 cbPassword = 0;
```

**Heap Inspection\Path 6:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=905 |
| Status | New |

Method rdp_write_info_packet at line 609 of FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c defines cbPassword, which is designated to contain user passwords. However, while plaintext passwords are later assigned to cbPassword, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c |
| Line | 618 | 618 |
| Object | cbPassword | cbPassword |

Code Snippet
File Name FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
618.        UINT16 cbPassword = 0;
```

**Heap Inspection\Path 7:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=906 |
| Status | New |

Method rdp_write_info_packet at line 602 of FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c defines cbPassword, which is designated to contain user passwords. However, while plaintext passwords are later assigned to cbPassword, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c |
| Line | 611 | 611 |
| Object | cbPassword | cbPassword |

Code Snippet
File Name FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
611.        UINT16 cbPassword = 0;
```

**Heap Inspection\Path 8:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=907 |
| Status | New |

Method rdp_write_info_packet at line 602 of FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c defines cbPassword, which is designated to contain user passwords. However, while plaintext passwords are later assigned to cbPassword, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c |
| Line | 611 | 611 |
| Object | cbPassword | cbPassword |

Code Snippet
File Name FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
611.        UINT16 cbPassword = 0;
```

**Heap Inspection\Path 9:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=908 |
| Status | New |

Method rdp_write_info_packet at line 721 of FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c defines passwordW, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwordW, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Line | 729 | 729 |
| Object | passwordW | passwordW |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s) |

```
....
729.         WCHAR* passwordW = NULL;
```

**Heap Inspection\Path 10:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=909 |
| Status | New |

Method rdp_write_info_packet at line 609 of FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c defines passwordW, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwordW, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Line | 617 | 617 |
| Object | passwordW | passwordW |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s) |

```
....
617.         WCHAR* passwordW = NULL;
```

**Heap Inspection\Path 11:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=910 |
| Status | New |

Method rdp_write_info_packet at line 609 of FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp defines passwordW, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwordW, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp | FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp |
| Line | 617 | 617 |
| Object | passwordW | passwordW |

Code Snippet
File Name FreeRDP@@FreeRDP-2.3.0-CVE-2024-32661-TP.cpp
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
617.        WCHAR* passwordW = NULL;
```

**Heap Inspection\Path 12:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=911 |
| Status | New |

Method rdp_write_info_packet at line 609 of FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c defines passwordW, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwordW, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c |
| Line | 617 | 617 |
| Object | passwordW | passwordW |

Code Snippet
File Name FreeRDP@@FreeRDP-2.4.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
617.        WCHAR* passwordW = NULL;
```

**Heap Inspection\Path 13:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=912 |
| Status | New |

Method rdp_write_info_packet at line 609 of FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c defines passwordW, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwordW, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c |
| Line | 617 | 617 |
| Object | passwordW | passwordW |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.5.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s) |

```
....
617.         WCHAR* passwordW = NULL;
```

**Heap Inspection\Path 14:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=913 |
| Status | New |

Method rdp_write_info_packet at line 609 of FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c defines passwordW, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwordW, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c |
| Line | 617 | 617 |
| Object | passwordW | passwordW |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.7.0-CVE-2024-32661-TP.c |
| Method | static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s) |

```
....
617.         WCHAR* passwordW = NULL;
```

**Heap Inspection\Path 15:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=914 |
| Status | New |

Method rdp_write_info_packet at line 602 of FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c defines passwordW, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwordW, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c |
| Line | 610 | 610 |
| Object | passwordW | passwordW |

Code Snippet
File Name FreeRDP@@FreeRDP-2.8.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
610.         WCHAR* passwordW = NULL;
```

### Heap Inspection\Path 16:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=915 |
| Status | New |

Method rdp_write_info_packet at line 602 of FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c defines passwordW, which is designated to contain user passwords. However, while plaintext passwords are later assigned to passwordW, this variable is never cleared from memory.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c | FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c |
| Line | 610 | 610 |
| Object | passwordW | passwordW |

Code Snippet
File Name FreeRDP@@FreeRDP-2.9.0-CVE-2024-32661-TP.c
Method static BOOL rdp_write_info_packet(rdpRdp* rdp, wStream* s)

```
....
610.         WCHAR* passwordW = NULL;
```

## Char Overflow

Query Path:
CPP\Cx\CPP Integer Overflow\Char Overflow Version:1

### Categories

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

*Description*

**Char Overflow\Path 1:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=369 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 486 of FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c |
| Line | 503 | 503 |
| Object | AssignExpr | AssignExpr |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c
Method static void fips_expand_key_bits(BYTE* in, BYTE* out)

```
....
503.                    out[i] = (buf[p] << r) & 0xfe;
```

**Char Overflow\Path 2:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=370 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 486 of FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c |
| Line | 508 | 508 |
| Object | AssignExpr | AssignExpr |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c
Method static void fips_expand_key_bits(BYTE* in, BYTE* out)

```
....
508.                    c = buf[p] << r;
```

**Char Overflow\Path 3:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=371 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 486 of FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c |
| Line | 509 | 509 |
| Object | AssignExpr | AssignExpr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c |
| Method | static void fips_expand_key_bits(BYTE* in, BYTE* out) |

```
....
509.                    c |= buf[p + 1] >> (8 - r);
```

**Char Overflow\Path 4:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=372 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 114 of FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c |
| Line | 155 | 155 |
| Object | AssignExpr | AssignExpr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2023-39354-TP.c |
| Method | static BOOL nsc_rle_decode(BYTE* in, BYTE* out, UINT32 outSize, UINT32 originalSize) |

```
....
155.                    out += len;
```

**Char Overflow\Path 5:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=373 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 114 of FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c |
| Line | 155 | 155 |
| Object | AssignExpr | AssignExpr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2023-39354-TP.c |
| Method | static BOOL nsc_rle_decode(BYTE* in, BYTE* out, UINT32 outSize, UINT32 originalSize) |

```
....
155.                    out += len;
```

**Char Overflow\Path 6:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=374 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 114 of FreeRDP@@FreeRDP-2.3.0-CVE-2023-39354-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2023-39354-TP.c |
| Line | 155 | 155 |
| Object | AssignExpr | AssignExpr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2023-39354-TP.c |
| Method | static BOOL nsc_rle_decode(BYTE* in, BYTE* out, UINT32 outSize, UINT32 originalSize) |

```
....
155.                    out += len;
```

**Char Overflow\Path 7:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=375 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 114 of FreeRDP@@FreeRDP-2.4.0-CVE-2023-39354-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.4.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.4.0-CVE-2023-39354-TP.c |
| Line | 155 | 155 |
| Object | AssignExpr | AssignExpr |

Code Snippet

File Name FreeRDP@@FreeRDP-2.4.0-CVE-2023-39354-TP.c

Method static BOOL nsc_rle_decode(BYTE* in, BYTE* out, UINT32 outSize, UINT32 originalSize)

```
....
155.                    out += len;
```

**Char Overflow\Path 8:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=376 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 114 of FreeRDP@@FreeRDP-2.5.0-CVE-2023-39354-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.5.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.5.0-CVE-2023-39354-TP.c |
| Line | 155 | 155 |
| Object | AssignExpr | AssignExpr |

Code Snippet

File Name FreeRDP@@FreeRDP-2.5.0-CVE-2023-39354-TP.c

Method static BOOL nsc_rle_decode(BYTE* in, BYTE* out, UINT32 outSize, UINT32 originalSize)

```
....
155.                    out += len;
```

**Char Overflow\Path 9:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=377 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 114 of FreeRDP@@FreeRDP-2.7.0-CVE-2023-39354-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.7.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.7.0-CVE-2023-39354-TP.c |
| Line | 155 | 155 |
| Object | AssignExpr | AssignExpr |

Code Snippet

File Name FreeRDP@@FreeRDP-2.7.0-CVE-2023-39354-TP.c

Method static BOOL nsc_rle_decode(BYTE* in, BYTE* out, UINT32 outSize, UINT32 originalSize)

```
....
155.                    out += len;
```

**Char Overflow\Path 10:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=378 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 114 of FreeRDP@@FreeRDP-2.8.0-CVE-2023-39354-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.8.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.8.0-CVE-2023-39354-TP.c |
| Line | 155 | 155 |
| Object | AssignExpr | AssignExpr |

Code Snippet

File Name FreeRDP@@FreeRDP-2.8.0-CVE-2023-39354-TP.c

Method static BOOL nsc_rle_decode(BYTE* in, BYTE* out, UINT32 outSize, UINT32 originalSize)

```
....
155.                        out += len;
```

### Char Overflow\Path 11:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=379 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 114 of FreeRDP@@FreeRDP-2.9.0-CVE-2023-39354-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.9.0-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-2.9.0-CVE-2023-39354-TP.c |
| Line | 155 | 155 |
| Object | AssignExpr | AssignExpr |

| | |
|---|---|
| Code Snippet | |
| File Name | FreeRDP@@FreeRDP-2.9.0-CVE-2023-39354-TP.c |
| Method | static BOOL nsc_rle_decode(BYTE* in, BYTE* out, UINT32 outSize, UINT32 originalSize) |

```
....
155.                        out += len;
```

### Char Overflow\Path 12:

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=380 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 115 of FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-39354-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-39354-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-39354-TP.c |
| Line | 156 | 156 |
| Object | AssignExpr | AssignExpr |

| | |
|---|---|
| Code Snippet | |
| File Name | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-39354-TP.c |

| | |
|---|---|
| Method | static BOOL nsc_rle_decode(BYTE* in, BYTE* out, UINT32 outSize, UINT32 originalSize) |

```
....
156.                        out += len;
```

## Divide By Zero

Query Path:

CPP\Cx\CPP Medium Threat\Divide By Zero Version:1

*Description*

**Divide By Zero\Path 1:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=13 |
| Status | New |

The application performs an illegal operation in xQueueGenericCreate, in FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c. In line 382, the program attempts to divide by uxItemSize, which might be evaluate to 0 (zero) at time of division. This value could be a hard-coded zero value, or received from external, untrusted input uxItemSize in xQueueGenericCreate of FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c, at line 382.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c |
| Line | 398 | 398 |
| Object | uxItemSize | uxItemSize |

Code Snippet

| | |
|---|---|
| File Name | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c |
| Method | QueueHandle_t xQueueGenericCreate( const UBaseType_t uxQueueLength, |

```
....
398.            configASSERT( ( uxItemSize == 0 ) || ( uxQueueLength == ( xQueueSizeInBytes / uxItemSize ) ) );
```

**Divide By Zero\Path 2:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=14 |
| Status | New |

The application performs an illegal operation in xQueueGenericCreate, in FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c. In line 382, the program attempts to divide by uxItemSize, which might be evaluate to 0 (zero) at time of division. This value could be a hard-coded zero value, or received from external, untrusted input uxItemSize in xQueueGenericCreate of FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c, at line 382.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c |
| Line | 398 | 398 |
| Object | uxItemSize | uxItemSize |

Code Snippet
File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c
Method QueueHandle_t xQueueGenericCreate( const UBaseType_t uxQueueLength,

```
....
398.            configASSERT( ( uxItemSize == 0 ) || ( uxQueueLength == ( 
xQueueSizeInBytes / uxItemSize ) ) );
```

**Divide By Zero\Path 3:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=15 |
| Status | New |

The application performs an illegal operation in xQueueGenericCreate, in FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c. In line 382, the program attempts to divide by uxItemSize, which might be evaluate to 0 (zero) at time of division. This value could be a hard-coded zero value, or received from external, untrusted input uxItemSize in xQueueGenericCreate of FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c, at line 382.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c |
| Line | 398 | 398 |
| Object | uxItemSize | uxItemSize |

Code Snippet
File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c
Method QueueHandle_t xQueueGenericCreate( const UBaseType_t uxQueueLength,

```
....
398.            configASSERT( ( uxItemSize == 0 ) || ( uxQueueLength == ( 
xQueueSizeInBytes / uxItemSize ) ) );
```

## 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=1000019&projectid=14&pathid=381 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 915 of FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Line | 1023 | 1023 |
| Object | AssignExpr | AssignExpr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Method | BOOL rdp_write_enhanced_security_redirection_packet(wStream* s, const rdpRedirection* redirection) |

```
....
1023.                   length += (UINT32)rcc;
```

**Integer Overflow\Path 2:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=382 |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 942 of FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Line | 1050 | 1050 |
| Object | AssignExpr | AssignExpr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Method | BOOL rdp_write_enhanced_security_redirection_packet(wStream* s, const rdpRedirection* redirection) |

```
....
1050.                    length +=  (UINT32)rcc;
```

**Integer Overflow\Path 3:**

| | |
|---|---|
| Severity | Medium |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=383](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=383) |
| Status | New |

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 930 of FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Line | 1038 | 1038 |
| Object | AssignExpr | AssignExpr |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Method | BOOL rdp_write_enhanced_security_redirection_packet(wStream* s, const rdpRedirection* redirection) |

```
....
1038.                    length +=  (UINT32)rcc;
```

## 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=1000019&projectid=14&pathid=1194](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1194) |
| Status | New |

The general_name_type_label method calls the sprintf function, at line 346 of FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-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 | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c |
| Line | 356 | 356 |
| Object | sprintf | sprintf |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c
Method static const char* general_name_type_label(int general_name_type)

```
....
356.                sprintf(buffer, "Unknown general name type (%d)",
general_name_type);
```

### Unchecked Return Value\Path 2:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1195 |
| Status | New |

The wf_cliprdr_get_file_descriptor method calls the wcscpy_s function, at line 1664 of FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-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 | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1691 | 1691 |
| Object | wcscpy_s | wcscpy_s |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method static FILEDESCRIPTORW* wf_cliprdr_get_file_descriptor(WCHAR* file_name, size_t pathLen)

```
....
1691.       wcscpy_s(fd->cFileName, sizeof(fd->cFileName) / 2, file_name
+ pathLen);
```

### Unchecked Return Value\Path 3:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1196 |
| Status | New |

The wf_cliprdr_add_to_file_arrays method calls the wcscpy_s function, at line 1727 of FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-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 | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1739 | 1739 |
| Object | wcscpy_s | wcscpy_s |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method static BOOL wf_cliprdr_add_to_file_arrays(wfClipboard* clipboard, WCHAR* full_file_name,

```
....
1739.       wcscpy_s(clipboard->file_names[clipboard->nFiles], MAX_PATH,
full_file_name);
```

**Unchecked Return Value\Path 4:**

Severity Low
Result State To Verify
Online Results http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1197
Status New

The wf_cliprdr_get_file_descriptor method calls the wcscpy_s function, at line 1665 of FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-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 | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1692 | 1692 |
| Object | wcscpy_s | wcscpy_s |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c
Method static FILEDESCRIPTORW* wf_cliprdr_get_file_descriptor(WCHAR* file_name, size_t pathLen)

```
....
1692.       wcscpy_s(fd->cFileName, sizeof(fd->cFileName) / 2, file_name
+ pathLen);
```

**Unchecked Return Value\Path 5:**

Severity Low
Result State To Verify

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1198 |
| Status | New |

The wf_cliprdr_add_to_file_arrays method calls the wcscpy_s function, at line 1728 of FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-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 | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1740 | 1740 |
| Object | wcscpy_s | wcscpy_s |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | static BOOL wf_cliprdr_add_to_file_arrays(wfClipboard* clipboard, WCHAR* full_file_name, |

```
....
1740.       wcscpy_s(clipboard->file_names[clipboard->nFiles], MAX_PATH,
full_file_name);
```

**Unchecked Return Value\Path 6:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1199 |
| Status | New |

The wf_cliprdr_get_file_descriptor method calls the wcscpy_s function, at line 1665 of FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-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 | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 1692 | 1692 |
| Object | wcscpy_s | wcscpy_s |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Method | static FILEDESCRIPTORW* wf_cliprdr_get_file_descriptor(WCHAR* file_name, size_t pathLen) |

```
....
1692.       wcscpy_s(fd->cFileName, sizeof(fd->cFileName) / 2, file_name
+ pathLen);
```

**Unchecked Return Value\Path 7:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1200](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1200) |
| Status | New |

The wf_cliprdr_add_to_file_arrays method calls the wcscpy_s function, at line 1728 of FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-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 | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 1740 | 1740 |
| Object | wcscpy_s | wcscpy_s |

| | |
|---|---|
| Code Snippet | |
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Method | static BOOL wf_cliprdr_add_to_file_arrays(wfClipboard* clipboard, WCHAR* full_file_name, |

```
....
1740.       wcscpy_s(clipboard->file_names[clipboard->nFiles], MAX_PATH,
full_file_name);
```

**Unchecked Return Value\Path 8:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1201](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1201) |
| Status | New |

The crypto_cert_get_public_key method calls the Pointer function, at line 53 of FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-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 | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c |
| Line | 78 | 78 |
| Object | Pointer | Pointer |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c |
| Method | BOOL crypto_cert_get_public_key(CryptoCert cert, BYTE** PublicKey, DWORD* PublicKeyLength) |

```
....
78.    *PublicKey = (BYTE*)malloc(length);
```

### Unchecked Return Value\Path 9:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1202 |
| Status | New |

The crypto_cert_get_dns_names method calls the Pointer function, at line 734 of FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-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 | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c |
| Line | 752 | 752 |
| Object | Pointer | Pointer |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c |
| Method | char** crypto_cert_get_dns_names(X509* x509, int* count, int** lengths) |

```
....
752.        (*lengths) = calloc(list.count, sizeof(**lengths));
```

### Unchecked Return Value\Path 10:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1203 |
| Status | New |

The wf_cliprdr_server_format_data_request method calls the wFileName function, at line 2040 of FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-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 | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 2124 | 2124 |

| | | |
|---|---|---|
| Object | wFileName | wFileName |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2124.                    wFileName = (LPWSTR)calloc(cchWideChar,
sizeof(WCHAR));
```

**Unchecked Return Value\Path 11:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1204 |
| Status | New |

The wf_cliprdr_server_format_data_request method calls the buff function, at line 2040 of FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-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 | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 2166 | 2166 |
| Object | buff | buff |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2166.                    buff = malloc(size);
```

**Unchecked Return Value\Path 12:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1205 |
| Status | New |

The wf_cliprdr_server_format_data_request method calls the wFileName function, at line 2041 of FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-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 | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |

| | | |
|---|---|---|
| Line | 2125 | 2125 |
| Object | wFileName | wFileName |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2125.                    wFileName = (LPWSTR)calloc(cchWideChar,
sizeof(WCHAR));
```

**Unchecked Return Value\Path 13:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1206 |
| Status | New |

The wf_cliprdr_server_format_data_request method calls the buff function, at line 2041 of FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-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 | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 2167 | 2167 |
| Object | buff | buff |

Code Snippet
File Name FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2167.                    buff = malloc(size);
```

**Unchecked Return Value\Path 14:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1207 |
| Status | New |

The wf_cliprdr_server_format_data_request method calls the wFileName function, at line 2041 of FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-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 | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 2125 | 2125 |
| Object | wFileName | wFileName |

Code Snippet
File Name FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2125.                   wFileName = (LPWSTR)calloc(cchWideChar,
sizeof(WCHAR));
```

**Unchecked Return Value\Path 15:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1208 |
| Status | New |

The wf_cliprdr_server_format_data_request method calls the buff function, at line 2041 of FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-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 | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 2167 | 2167 |
| Object | buff | buff |

Code Snippet
File Name FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c
Method wf_cliprdr_server_format_data_request(CliprdrClientContext* context,

```
....
2167.                   buff = malloc(size);
```

**Unchecked Return Value\Path 16:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1209 |
| Status | New |

The redirection_copy_data method calls the Pointer function, at line 101 of FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-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 | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Line | 110 | 110 |
| Object | Pointer | Pointer |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c
Method static BOOL redirection_copy_data(BYTE** dst, UINT32* plen, const BYTE* str, size_t len)

```
....
110.         *dst = malloc(len);
```

**Unchecked Return Value\Path 17:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1210 |
| Status | New |

The redirection_copy_array method calls the Pointer function, at line 118 of FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-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 | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Line | 125 | 125 |
| Object | Pointer | Pointer |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c
Method static BOOL redirection_copy_array(char*** dst, UINT32* plen, const char** str, size_t len)

```
....
125.         *dst = calloc(len, sizeof(char));
```

**Unchecked Return Value\Path 18:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1211 |
| Status | New |

The rdp_redirection_read_base64_wchar method calls the Pointer function, at line 243 of FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-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 | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Line | 262 | 262 |
| Object | Pointer | Pointer |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Method | static BOOL rdp_redirection_read_base64_wchar(UINT32 flag, wStream* s, UINT32* pLength, |

```
....
262.          *pData = calloc(utf8_len, sizeof(BYTE));
```

**Unchecked Return Value\Path 19:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1212 |
| Status | New |

The rdp_redirection_read_data method calls the Pointer function, at line 623 of FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-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 | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Line | 632 | 632 |
| Object | Pointer | Pointer |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Method | static BOOL rdp_redirection_read_data(UINT32 flag, wStream* s, UINT32* pLength, BYTE** pData) |

```
....
632.          *pData = (BYTE*)malloc(*pLength);
```

**Unchecked Return Value\Path 20:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14 |

| | |
|---|---|
| | &pathid=1213 |
| Status | New |

The redirection_copy_data method calls the Pointer function, at line 103 of FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-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 | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Line | 112 | 112 |
| Object | Pointer | Pointer |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c
Method static BOOL redirection_copy_data(BYTE** dst, UINT32* plen, const BYTE* str, size_t len)

```
....
112.         *dst = malloc(len);
```

**Unchecked Return Value\Path 21:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1214 |
| Status | New |

The redirection_copy_array method calls the Pointer function, at line 120 of FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-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 | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Line | 127 | 127 |
| Object | Pointer | Pointer |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c
Method static BOOL redirection_copy_array(char*** dst, UINT32* plen, const char** str, size_t len)

```
....
127.         *dst = calloc(len, sizeof(char));
```

**Unchecked Return Value\Path 22:**

| | |
|---|---|
| Severity | Low |

| | |
|---|---|
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1215 |
| Status | New |

The rdp_redirection_read_base64_wchar method calls the Pointer function, at line 245 of FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-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 | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Line | 264 | 264 |
| Object | Pointer | Pointer |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Method | static BOOL rdp_redirection_read_base64_wchar(UINT32 flag, wStream* s, UINT32* pLength, |

```
....
264.        *pData = calloc(utf8_len, sizeof(BYTE));
```

**Unchecked Return Value\Path 23:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1216 |
| Status | New |

The rdp_redirection_read_data method calls the Pointer function, at line 637 of FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-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 | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Line | 646 | 646 |
| Object | Pointer | Pointer |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Method | static BOOL rdp_redirection_read_data(UINT32 flag, wStream* s, UINT32* pLength, BYTE** pData) |

```
....
646.        *pData = (BYTE*)malloc(*pLength);
```

### Unchecked Return Value\Path 24:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1217 |
| Status | New |

The redirection_copy_data method calls the Pointer function, at line 103 of FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-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 | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Line | 112 | 112 |
| Object | Pointer | Pointer |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Method | static BOOL redirection_copy_data(BYTE** dst, UINT32* plen, const BYTE* str, size_t len) |

```
....
112.        *dst = malloc(len);
```

### Unchecked Return Value\Path 25:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1218 |
| Status | New |

The redirection_copy_array method calls the Pointer function, at line 120 of FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-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 | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Line | 127 | 127 |
| Object | Pointer | Pointer |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Method | static BOOL redirection_copy_array(char*** dst, UINT32* plen, const char** str, size_t len) |

```
....
127.        *dst = calloc(len, sizeof(char*));
```

**Unchecked Return Value\Path 26:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1219](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1219) |
| Status | New |

The rdp_redirection_read_base64_wchar method calls the Pointer function, at line 234 of FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-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 | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Line | 253 | 253 |
| Object | Pointer | Pointer |

| Code Snippet | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Method | static BOOL rdp_redirection_read_base64_wchar(UINT32 flag, wStream* s, UINT32* pLength, |

```
....
253.        *pData = calloc(utf8_len, sizeof(BYTE));
```

**Unchecked Return Value\Path 27:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1220](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1220) |
| Status | New |

The rdp_redirection_read_data method calls the Pointer function, at line 625 of FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-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 | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Line | 634 | 634 |
| Object | Pointer | Pointer |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Method | static BOOL rdp_redirection_read_data(UINT32 flag, wStream* s, UINT32* pLength, BYTE** pData) |

```
....
634.         *pData = (BYTE*)malloc(*pLength);
```

### Unchecked Return Value\Path 28:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1221 |
| Status | New |

The stun_parse_attribute method calls the data function, at line 114 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 181 | 181 |
| Object | data | data |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Method | int stun_parse_attribute(stun_msg_t *msg, unsigned char *p) |

```
....
181.       attr->enc_buf.data = (unsigned char *) malloc(len);
```

### Unchecked Return Value\Path 29:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1222 |
| Status | New |

The stun_parse_attr_address method calls the addr function, at line 200 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 213 | 213 |
| Object | addr | addr |

Code Snippet

|  |  |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Method | int stun_parse_attr_address(stun_attr_t *attr, |

```
....
213.     addr = (su_sockaddr_t *) malloc(addrlen);
```

### Unchecked Return Value\Path 30:

|  |  |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1223](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1223) |
| Status | New |

The stun_parse_attr_error_code method calls the error function, at line 235 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 242 | 242 |
| Object | error | error |

Code Snippet

|  |  |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Method | int stun_parse_attr_error_code(stun_attr_t *attr, const unsigned char *p, unsigned len) { |

```
....
242.     error = (stun_attr_errorcode_t *) malloc(sizeof(*error));
```

### Unchecked Return Value\Path 31:

|  |  |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1224](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1224) |
| Status | New |

The stun_parse_attr_error_code method calls the phrase function, at line 235 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 246 | 246 |

| | | |
|---|---|---|
| Object | phrase | phrase |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_parse_attr_error_code(stun_attr_t *attr, const unsigned char *p, unsigned len) {

```
....
246.     error->phrase = (char *) malloc(len-3);
```

**Unchecked Return Value\Path 32:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1225](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1225) |
| Status | New |

The stun_parse_attr_uint32 method calls the cr function, at line 257 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 261 | 261 |
| Object | cr | cr |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_parse_attr_uint32(stun_attr_t *attr, const unsigned char *p, unsigned len)

```
....
261.     cr = (stun_attr_changerequest_t *) malloc(sizeof(*cr));
```

**Unchecked Return Value\Path 33:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1226](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1226) |
| Status | New |

The stun_parse_attr_buffer method calls the buf function, at line 270 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 273 | 273 |
| Object | buf | buf |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_parse_attr_buffer(stun_attr_t *attr, const unsigned char *p, unsigned len)

```
....
273.    buf = (stun_buffer_t *) malloc(sizeof(stun_buffer_t));
```

**Unchecked Return Value\Path 34:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1227 |
| Status | New |

The stun_encode_message_integrity method calls the padded_text function, at line 434 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 451 | 451 |
| Object | padded_text | padded_text |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_encode_message_integrity(stun_attr_t *attr,

```
....
451.      padded_text = (unsigned char *) malloc(padded_len);
```

**Unchecked Return Value\Path 35:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1228 |
| Status | New |

The stun_encode_type_len method calls the data function, at line 478 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 481 | 481 |
| Object | data | data |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_encode_type_len(stun_attr_t *attr, uint16_t len) {

```
....
481.    attr->enc_buf.data = (unsigned char *) malloc(len + 4);
```

**Unchecked Return Value\Path 36:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1229 |
| Status | New |

The stun_validate_message_integrity method calls the padded_text function, at line 499 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 527 | 527 |
| Object | padded_text | padded_text |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_validate_message_integrity(stun_msg_t *msg, stun_buffer_t *pwd)

```
....
527.    padded_text = (unsigned char *) malloc(padded_len);
```

**Unchecked Return Value\Path 37:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1230 |
| Status | New |

The *stun_determine_ip_address method calls the local_ip_address function, at line 763 of freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 786 | 786 |
| Object | local_ip_address | local_ip_address |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method char *stun_determine_ip_address(int family)

```
....
786.     local_ip_address = malloc(address_size + 1);
```

**Unchecked Return Value\Path 38:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1231 |
| Status | New |

The stun_parse_attribute method calls the data function, at line 114 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 181 | 181 |
| Object | data | data |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_parse_attribute(stun_msg_t *msg, unsigned char *p)

```
....
181.       attr->enc_buf.data = (unsigned char *) malloc(len);
```

**Unchecked Return Value\Path 39:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1232 |
| Status | New |

The stun_parse_attr_address method calls the addr function, at line 200 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 213 | 213 |
| Object | addr | addr |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_parse_attr_address(stun_attr_t *attr,

```
....
213.     addr = (su_sockaddr_t *) malloc(addrlen);
```

**Unchecked Return Value\Path 40:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1233 |
| Status | New |

The stun_parse_attr_error_code method calls the error function, at line 235 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 242 | 242 |
| Object | error | error |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_parse_attr_error_code(stun_attr_t *attr, const unsigned char *p, unsigned len) {

```
....
242.     error = (stun_attr_errorcode_t *) malloc(sizeof(*error));
```

**Unchecked Return Value\Path 41:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1234 |
| Status | New |

The stun_parse_attr_error_code method calls the phrase function, at line 235 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 246 | 246 |
| Object | phrase | phrase |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_parse_attr_error_code(stun_attr_t *attr, const unsigned char *p, unsigned len) {

```
....
246.     error->phrase = (char *) malloc(len-3);
```

**Unchecked Return Value\Path 42:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1235 |
| Status | New |

The stun_parse_attr_uint32 method calls the cr function, at line 257 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 261 | 261 |
| Object | cr | cr |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_parse_attr_uint32(stun_attr_t *attr, const unsigned char *p, unsigned len)

```
....
261.     cr = (stun_attr_changerequest_t *) malloc(sizeof(*cr));
```

**Unchecked Return Value\Path 43:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14 |

| | |
|---|---|
| | &pathid=1236 |
| Status | New |

The stun_parse_attr_buffer method calls the buf function, at line 270 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 273 | 273 |
| Object | buf | buf |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Method | int stun_parse_attr_buffer(stun_attr_t *attr, const unsigned char *p, unsigned len) |

```
....
273.     buf = (stun_buffer_t *) malloc(sizeof(stun_buffer_t));
```

**Unchecked Return Value\Path 44:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1237 |
| Status | New |

The stun_encode_message_integrity method calls the padded_text function, at line 434 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 451 | 451 |
| Object | padded_text | padded_text |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Method | int stun_encode_message_integrity(stun_attr_t *attr, |

```
....
451.       padded_text = (unsigned char *) malloc(padded_len);
```

**Unchecked Return Value\Path 45:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |

| | |
|---|---|
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1238 |
| Status | New |

The stun_encode_type_len method calls the data function, at line 478 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 481 | 481 |
| Object | data | data |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_encode_type_len(stun_attr_t *attr, uint16_t len) {

```
....
481.     attr->enc_buf.data = (unsigned char *) malloc(len + 4);
```

**Unchecked Return Value\Path 46:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1239 |
| Status | New |

The stun_validate_message_integrity method calls the padded_text function, at line 499 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 527 | 527 |
| Object | padded_text | padded_text |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_validate_message_integrity(stun_msg_t *msg, stun_buffer_t *pwd)

```
....
527.     padded_text = (unsigned char *) malloc(padded_len);
```

**Unchecked Return Value\Path 47:**

| | |
|---|---|
| Severity | Low |

| | |
|---|---|
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1240 |
| Status | New |

The *stun_determine_ip_address method calls the local_ip_address function, at line 763 of freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 786 | 786 |
| Object | local_ip_address | local_ip_address |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Method | char *stun_determine_ip_address(int family) |

```
....
786.     local_ip_address = malloc(address_size + 1);
```

**Unchecked Return Value\Path 48:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1241 |
| Status | New |

The stun_parse_attribute method calls the data function, at line 114 of freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 181 | 181 |
| Object | data | data |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Method | int stun_parse_attribute(stun_msg_t *msg, unsigned char *p) |

```
....
181.       attr->enc_buf.data = (unsigned char *) malloc(len);
```

**Unchecked Return Value\Path 49:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1242 |
| Status | New |

The stun_parse_attr_address method calls the addr function, at line 200 of freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 213 | 213 |
| Object | addr | addr |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c
Method int stun_parse_attr_address(stun_attr_t *attr,

```
....
213.     addr = (su_sockaddr_t *) malloc(addrlen);
```

### Unchecked Return Value\Path 50:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1243 |
| Status | New |

The stun_parse_attr_error_code method calls the error function, at line 235 of freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-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 | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 242 | 242 |
| Object | error | error |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c
Method int stun_parse_attr_error_code(stun_attr_t *attr, const unsigned char *p, unsigned len) {

```
....
242.     error = (stun_attr_errorcode_t *) malloc(sizeof(*error));
```

## 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=1000019&projectid=14&pathid=1315 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c |
| Line | 503 | 503 |
| Object | p | p |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c |
| Method | static void fips_expand_key_bits(BYTE* in, BYTE* out) |

```
....
503.                    out[i] = (buf[p] << r) & 0xfe;
```

**Unchecked Array Index\Path 2:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1316 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c |
| Line | 508 | 508 |
| Object | p | p |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2020-13397-TP.c |
| Method | static void fips_expand_key_bits(BYTE* in, BYTE* out) |

```
....
508.                    c = buf[p] << r;
```

### Unchecked Array Index\Path 3:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1317 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c |
| Line | 592 | 592 |
| Object | i | i |

Code Snippet

File Name FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c

Method static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context)

```
....
592.                  subSystems[i] = &g_Subsystem_mediacodec;
```

### Unchecked Array Index\Path 4:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1318 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c |
| Line | 598 | 598 |
| Object | i | i |

Code Snippet

File Name FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c

Method static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context)

```
....
598.                  subSystems[i] = &g_Subsystem_MF;
```

**Unchecked Array Index\Path 5:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1319 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c |
| Line | 604 | 604 |
| Object | i | i |

Code Snippet

File Name FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c

Method static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context)

```
....
604.                subSystems[i] = &g_Subsystem_OpenH264;
```

**Unchecked Array Index\Path 6:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1320 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c |
| Line | 610 | 610 |
| Object | i | i |

Code Snippet

File Name FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2023-40187-FP.c

Method static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context)

```
....
610.                subSystems[i] = &g_Subsystem_libavcodec;
```

**Unchecked Array Index\Path 7:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14 |

| | |
|---|---|
| | &pathid=1321 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c |
| Line | 602 | 602 |
| Object | i | i |

Code Snippet

File Name FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c

Method static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context)

```
....
602.            subSystems[i] = &g_Subsystem_mediacodec;
```

**Unchecked Array Index\Path 8:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1322 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c |
| Line | 608 | 608 |
| Object | i | i |

Code Snippet

File Name FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c

Method static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context)

```
....
608.            subSystems[i] = &g_Subsystem_MF;
```

**Unchecked Array Index\Path 9:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1323 |
| Status | New |

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

| | | |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c |
| Line | 614 | 614 |
| Object | i | i |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c
Method static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context)

```
....
614.            subSystems[i] = &g_Subsystem_OpenH264;
```

**Unchecked Array Index\Path 10:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1324 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c |
| Line | 620 | 620 |
| Object | i | i |

Code Snippet
File Name FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2023-40187-FP.c
Method static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context)

```
....
620.            subSystems[i] = &g_Subsystem_libavcodec;
```

**Unchecked Array Index\Path 11:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1325 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c |
| Line | 604 | 604 |

| | | |
|---|---|---|
| Object | i | i |

Code Snippet
File Name FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c
Method static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context)

```
....
604.                subSystems[i] = &g_Subsystem_mediacodec;
```

**Unchecked Array Index\Path 12:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1326 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c |
| Line | 610 | 610 |
| Object | i | i |

Code Snippet
File Name FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c
Method static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context)

```
....
610.                subSystems[i] = &g_Subsystem_MF;
```

**Unchecked Array Index\Path 13:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1327 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c |
| Line | 616 | 616 |
| Object | i | i |

Code Snippet
File Name FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c

| | |
|---|---|
| Method | static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context) |

```
....
616.                subSystems[i] = &g_Subsystem_OpenH264;
```

**Unchecked Array Index\Path 14:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1328 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c |
| Line | 622 | 622 |
| Object | i | i |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.4.0-CVE-2023-40187-FP.c |
| Method | static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context) |

```
....
622.                subSystems[i] = &g_Subsystem_libavcodec;
```

**Unchecked Array Index\Path 15:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1329 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c |
| Line | 642 | 642 |
| Object | i | i |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c |
| Method | static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context) |

```
....
642.                subSystems[i] = &g_Subsystem_mediacodec;
```

**Unchecked Array Index\Path 16:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1330 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c |
| Line | 648 | 648 |
| Object | i | i |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c |
| Method | static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context) |

```
....
648.                subSystems[i] = &g_Subsystem_MF;
```

**Unchecked Array Index\Path 17:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1331 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c |
| Line | 654 | 654 |
| Object | i | i |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c |
| Method | static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context) |

```
....
654.                subSystems[i] = &g_Subsystem_OpenH264;
```

**Unchecked Array Index\Path 18:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1332 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c | FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c |
| Line | 660 | 660 |
| Object | i | i |

Code Snippet
File Name FreeRDP@@FreeRDP-3.6.0-CVE-2023-40187-FP.c
Method static BOOL CALLBACK h264_register_subsystems(PINIT_ONCE once, PVOID param, PVOID* context)

```
....
660.                subSystems[i] = &g_Subsystem_libavcodec;
```

**Unchecked Array Index\Path 19:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1333 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 90 | 90 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
90.    msg->stun_hdr.msg_type = get16(p, 0);
```

**Unchecked Array Index\Path 20:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1334 |

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

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 90 | 90 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
90.     msg->stun_hdr.msg_type = get16(p, 0);
```

**Unchecked Array Index\Path 21:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1335 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 91 | 91 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
91.     msg->stun_hdr.msg_len = get16(p, 2);
```

**Unchecked Array Index\Path 22:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1336 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |

| | | |
|---|---|---|
| Line | 91 | 91 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
91.     msg->stun_hdr.msg_len = get16(p, 2);
```

### Unchecked Array Index\Path 23:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1337 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 90 | 90 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
90.     msg->stun_hdr.msg_type = get16(p, 0);
```

### Unchecked Array Index\Path 24:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1338 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 90 | 90 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c

| | |
|---|---|
| Method | int stun_parse_message(stun_msg_t *msg) |

```
....
90.     msg->stun_hdr.msg_type = get16(p, 0);
```

**Unchecked Array Index\Path 25:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1339 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 91 | 91 |
| Object | p | p |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Method | int stun_parse_message(stun_msg_t *msg) |

```
....
91.     msg->stun_hdr.msg_len = get16(p, 2);
```

**Unchecked Array Index\Path 26:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1340 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 91 | 91 |
| Object | p | p |

Code Snippet

| | |
|---|---|
| File Name | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Method | int stun_parse_message(stun_msg_t *msg) |

```
....
91.     msg->stun_hdr.msg_len = get16(p, 2);
```

**Unchecked Array Index\Path 27:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1341 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 90 | 90 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
90.     msg->stun_hdr.msg_type = get16(p, 0);
```

**Unchecked Array Index\Path 28:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1342 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 90 | 90 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
90.     msg->stun_hdr.msg_type = get16(p, 0);
```

**Unchecked Array Index\Path 29:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1343 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 91 | 91 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
91.     msg->stun_hdr.msg_len = get16(p, 2);
```

**Unchecked Array Index\Path 30:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1344](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1344) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 91 | 91 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
91.     msg->stun_hdr.msg_len = get16(p, 2);
```

**Unchecked Array Index\Path 31:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1345](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1345) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c |
| Line | 90 | 90 |

| | | |
|---|---|---|
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
90.    msg->stun_hdr.msg_type = get16(p, 0);
```

**Unchecked Array Index\Path 32:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1346 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c |
| Line | 90 | 90 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
90.    msg->stun_hdr.msg_type = get16(p, 0);
```

**Unchecked Array Index\Path 33:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1347 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c |
| Line | 91 | 91 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
91.     msg->stun_hdr.msg_len = get16(p, 2);
```

**Unchecked Array Index\Path 34:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1348](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1348) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c |
| Line | 91 | 91 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
91.     msg->stun_hdr.msg_len = get16(p, 2);
```

**Unchecked Array Index\Path 35:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1349](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1349) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c |
| Line | 90 | 90 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
90.     msg->stun_hdr.msg_type = get16(p, 0);
```

**Unchecked Array Index\Path 36:**

| | |
|---|---|
| Severity | Low |

| | |
|---|---|
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1350 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c |
| Line | 90 | 90 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
90.     msg->stun_hdr.msg_type = get16(p, 0);
```

**Unchecked Array Index\Path 37:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1351 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c |
| Line | 91 | 91 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
91.     msg->stun_hdr.msg_len = get16(p, 2);
```

**Unchecked Array Index\Path 38:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1352 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c |
| Line | 91 | 91 |
| Object | p | p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c
Method int stun_parse_message(stun_msg_t *msg)

```
....
91.     msg->stun_hdr.msg_len = get16(p, 2);
```

## 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=1000019&projectid=14&pathid=1282 |
| Status | New |

The variable declared in null at FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c in line 1363 is not initialized when it is used by clipboard at FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c in line 1363.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1365 | 1385 |
| Object | null | clipboard |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method static LRESULT CALLBACK cliprdr_proc(HWND hWnd, UINT Msg, WPARAM wParam, LPARAM lParam)

```
....
1365.       static wfClipboard* clipboard = NULL;
....
1385.                           clipboard-
>RemoveClipboardFormatListener(hWnd);
```

### NULL Pointer Dereference\Path 2:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1283](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1283) |
| Status | New |

The variable declared in null at FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c in line 1364 is not initialized when it is used by clipboard at FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c in line 1364.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1366 | 1386 |
| Object | null | clipboard |

| | |
|---|---|
| Code Snippet | |
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | static LRESULT CALLBACK cliprdr_proc(HWND hWnd, UINT Msg, WPARAM wParam, LPARAM lParam) |

```
....
1366.        static wfClipboard* clipboard = NULL;
....
1386.                          clipboard-
>RemoveClipboardFormatListener(hWnd);
```

### NULL Pointer Dereference\Path 3:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1284](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1284) |
| Status | New |

The variable declared in null at FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c in line 1364 is not initialized when it is used by clipboard at FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c in line 1364.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 1366 | 1386 |
| Object | null | clipboard |

| | |
|---|---|
| Code Snippet | |
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Method | static LRESULT CALLBACK cliprdr_proc(HWND hWnd, UINT Msg, WPARAM wParam, LPARAM lParam) |

```
....
1366.       static wfClipboard* clipboard = NULL;
....
1386.                             clipboard-
>RemoveClipboardFormatListener(hWnd);
```

**NULL Pointer Dereference\Path 4:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1285 |
| Status | New |

The variable declared in null at FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32659-TP.c in line 985 is not initialized when it is used by palette at FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32659-TP.c in line 985.

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32659-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32659-TP.c |
| Line | 1313 | 1313 |
| Object | null | palette |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32659-TP.c |
| Method | void FreeRDPSplitColor(UINT32 color, UINT32 format, BYTE* _r, BYTE* _g, BYTE* _b, BYTE* _a, |

```
....
1313.                             FreeRDPSplitColor(tmp, palette->format,
_r, _g, _b, _a, NULL);
```

**NULL Pointer Dereference\Path 5:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1286 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c in line 2933 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c in line 2206.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c |
| Line | 2936 | 2222 |
| Object | null | u |

Code Snippet

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c

Method QueueSetMemberHandle_t xQueueSelectFromSet( QueueSetHandle_t xQueueSet,

```
....
2936.         QueueSetMemberHandle_t xReturn = NULL;
```

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c

Method static void prvCopyDataFromQueue( Queue_t * const pxQueue,

```
....
2222.           ( void ) memcpy( ( void * ) pvBuffer, ( void * ) pxQueue->u.xQueue.pcReadFrom, ( size_t ) pxQueue->uxItemSize ); /*lint !e961 !e418 !e9087 MISRA exception as the casts are only redundant for some ports.  Also previous logic ensures a null pointer can only be passed to memcpy() when the count is 0.  Cast to void required by function signature and safe as no alignment requirement and copy length specified in bytes. */
```

**NULL Pointer Dereference\Path 6:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1287 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c in line 2947 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c in line 2206.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c |
| Line | 2949 | 2222 |
| Object | null | u |

Code Snippet

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c

Method QueueSetMemberHandle_t xQueueSelectFromSetFromISR( QueueSetHandle_t xQueueSet )

```
....
2949.         QueueSetMemberHandle_t xReturn = NULL;
```

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c

Method static void prvCopyDataFromQueue( Queue_t * const pxQueue,

```
....
2222.           ( void ) memcpy( ( void * ) pvBuffer, ( void * ) pxQueue->u.xQueue.pcReadFrom, ( size_t ) pxQueue->uxItemSize ); /*lint !e961 !e418 !e9087 MISRA exception as the casts are only redundant for some ports.  Also previous logic ensures a null pointer can only be passed to memcpy() when the count is 0.  Cast to void required by function signature and safe as no alignment requirement and copy length specified in bytes. */
```

### NULL Pointer Dereference\Path 7:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1288 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c in line 2933 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c in line 2206.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c |
| Line | 2936 | 2213 |
| Object | null | u |

Code Snippet

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c

Method QueueSetMemberHandle_t xQueueSelectFromSet( QueueSetHandle_t xQueueSet,

```
....
2936.         QueueSetMemberHandle_t xReturn = NULL;
```

▼

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c

Method static void prvCopyDataFromQueue( Queue_t * const pxQueue,

```
....
2213.         if( pxQueue->u.xQueue.pcReadFrom >= pxQueue->u.xQueue.pcTail ) /*lint !e946 MISRA exception justified as use of the relational operator is the cleanest solutions. */
```

### NULL Pointer Dereference\Path 8:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1289 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c in line 2947 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c in line 2206.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c |
| Line | 2949 | 2213 |
| Object | null | u |

Code Snippet
File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c
Method QueueSetMemberHandle_t xQueueSelectFromSetFromISR( QueueSetHandle_t xQueueSet )

```
....
2949.           QueueSetMemberHandle_t xReturn = NULL;
```

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c
Method static void prvCopyDataFromQueue( Queue_t * const pxQueue,

```
....
2213.           if( pxQueue->u.xQueue.pcReadFrom >= pxQueue->u.xQueue.pcTail ) /*lint !e946 MISRA exception justified as use of the relational operator is the cleanest solutions. */
```

**NULL Pointer Dereference\Path 9:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1290 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c in line 2933 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c in line 2206.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c |
| Line | 2936 | 2213 |
| Object | null | u |

Code Snippet
File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c
Method QueueSetMemberHandle_t xQueueSelectFromSet( QueueSetHandle_t xQueueSet,

```
....
2936.          QueueSetMemberHandle_t xReturn = NULL;
```

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c

Method static void prvCopyDataFromQueue( Queue_t * const pxQueue,

```
....
2213.          if( pxQueue->u.xQueue.pcReadFrom >= pxQueue-
>u.xQueue.pcTail ) /*lint !e946 MISRA exception justified as use of the
relational operator is the cleanest solutions. */
```

**NULL Pointer Dereference\Path 10:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1291 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c in line 2947 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c in line 2206.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c |
| Line | 2949 | 2213 |
| Object | null | u |

Code Snippet

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c

Method QueueSetMemberHandle_t xQueueSelectFromSetFromISR( QueueSetHandle_t xQueueSet )

```
....
2949.          QueueSetMemberHandle_t xReturn = NULL;
```

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c

Method static void prvCopyDataFromQueue( Queue_t * const pxQueue,

```
....
2213.          if( pxQueue->u.xQueue.pcReadFrom >= pxQueue-
>u.xQueue.pcTail ) /*lint !e946 MISRA exception justified as use of the
relational operator is the cleanest solutions. */
```

**NULL Pointer Dereference\Path 11:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1292 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c in line 2933 is not initialized when it is used by pxQueue at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c in line 2347.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c |
| Line | 2936 | 2353 |
| Object | null | pxQueue |

Code Snippet

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c

Method QueueSetMemberHandle_t xQueueSelectFromSet( QueueSetHandle_t xQueueSet,

```
....
2936.          QueueSetMemberHandle_t xReturn = NULL;
```

▼

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-31571-TP.c

Method static BaseType_t prvIsQueueEmpty( const Queue_t * pxQueue )

```
....
2353.          if( pxQueue->uxMessagesWaiting == ( UBaseType_t ) 0 )
```

**NULL Pointer Dereference\Path 12:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1293 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c in line 2936 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c in line 2209.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c |
| Line | 2939 | 2225 |
| Object | null | u |

Code Snippet

| | |
|---|---|
| File Name | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c |
| Method | QueueSetMemberHandle_t xQueueSelectFromSet( QueueSetHandle_t xQueueSet, |

```
....
2939.           QueueSetMemberHandle_t xReturn = NULL;
```

▼

| | |
|---|---|
| File Name | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c |
| Method | static void prvCopyDataFromQueue( Queue_t * const pxQueue, |

```
....
2225.           ( void ) memcpy( ( void * ) pvBuffer, ( void * ) pxQueue-
>u.xQueue.pcReadFrom, ( size_t ) pxQueue->uxItemSize ); /*lint !e961
!e418 !e9087 MISRA exception as the casts are only redundant for some
ports.  Also previous logic ensures a null pointer can only be passed to
memcpy() when the count is 0.  Cast to void required by function
signature and safe as no alignment requirement and copy length specified
in bytes. */
```

**NULL Pointer Dereference\Path 13:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1294 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c in line 2950 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c in line 2209.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c |
| Line | 2952 | 2225 |
| Object | null | u |

Code Snippet

| | |
|---|---|
| File Name | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c |
| Method | QueueSetMemberHandle_t xQueueSelectFromSetFromISR( QueueSetHandle_t xQueueSet ) |

```
....
2952.           QueueSetMemberHandle_t xReturn = NULL;
```

▼

| | |
|---|---|
| File Name | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c |
| Method | static void prvCopyDataFromQueue( Queue_t * const pxQueue, |

```
....
2225.           ( void ) memcpy( ( void * ) pvBuffer, ( void * ) pxQueue->u.xQueue.pcReadFrom, ( size_t ) pxQueue->uxItemSize ); /*lint !e961 !e418 !e9087 MISRA exception as the casts are only redundant for some ports.  Also previous logic ensures a null pointer can only be passed to memcpy() when the count is 0.  Cast to void required by function signature and safe as no alignment requirement and copy length specified in bytes. */
```

### NULL Pointer Dereference\Path 14:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1295](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1295) |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c in line 2936 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c in line 2209.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c |
| Line | 2939 | 2216 |
| Object | null | u |

Code Snippet

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c

Method QueueSetMemberHandle_t xQueueSelectFromSet( QueueSetHandle_t xQueueSet,

```
....
2939.           QueueSetMemberHandle_t xReturn = NULL;
```

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c

Method static void prvCopyDataFromQueue( Queue_t * const pxQueue,

```
....
2216.           if( pxQueue->u.xQueue.pcReadFrom >= pxQueue->u.xQueue.pcTail ) /*lint !e946 MISRA exception justified as use of the relational operator is the cleanest solutions. */
```

### NULL Pointer Dereference\Path 15:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1296](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1296) |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c in line 2950 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c in line 2209.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c |
| Line | 2952 | 2216 |
| Object | null | u |

Code Snippet

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c

Method QueueSetMemberHandle_t xQueueSelectFromSetFromISR( QueueSetHandle_t xQueueSet )

```
....
2952.          QueueSetMemberHandle_t xReturn = NULL;
```

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c

Method static void prvCopyDataFromQueue( Queue_t * const pxQueue,

```
....
2216.          if( pxQueue->u.xQueue.pcReadFrom >= pxQueue-
>u.xQueue.pcTail ) /*lint !e946 MISRA exception justified as use of the
relational operator is the cleanest solutions. */
```

**NULL Pointer Dereference\Path 16:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1297 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c in line 2936 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c in line 2209.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c |
| Line | 2939 | 2216 |
| Object | null | u |

Code Snippet

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c

Method QueueSetMemberHandle_t xQueueSelectFromSet( QueueSetHandle_t xQueueSet,

```
....
2939.           QueueSetMemberHandle_t xReturn = NULL;
```

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c

Method static void prvCopyDataFromQueue( Queue_t * const pxQueue,

```
....
2216.           if( pxQueue->u.xQueue.pcReadFrom >= pxQueue-
>u.xQueue.pcTail ) /*lint !e946 MISRA exception justified as use of the
relational operator is the cleanest solutions. */
```

### NULL Pointer Dereference\Path 17:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1298 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c in line 2950 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c in line 2209.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c |
| Line | 2952 | 2216 |
| Object | null | u |

Code Snippet

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c

Method QueueSetMemberHandle_t xQueueSelectFromSetFromISR( QueueSetHandle_t xQueueSet )

```
....
2952.           QueueSetMemberHandle_t xReturn = NULL;
```

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c

Method static void prvCopyDataFromQueue( Queue_t * const pxQueue,

```
....
2216.           if( pxQueue->u.xQueue.pcReadFrom >= pxQueue-
>u.xQueue.pcTail ) /*lint !e946 MISRA exception justified as use of the
relational operator is the cleanest solutions. */
```

### NULL Pointer Dereference\Path 18:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1299 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c in line 2936 is not initialized when it is used by pxQueue at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c in line 2350.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c |
| Line | 2939 | 2356 |
| Object | null | pxQueue |

Code Snippet

| | |
|---|---|
| File Name | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c |
| Method | QueueSetMemberHandle_t xQueueSelectFromSet( QueueSetHandle_t xQueueSet, |

```
....
2939.           QueueSetMemberHandle_t xReturn = NULL;
```

| | |
|---|---|
| File Name | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-31571-TP.c |
| Method | static BaseType_t prvIsQueueEmpty( const Queue_t * pxQueue ) |

```
....
2356.           if( pxQueue->uxMessagesWaiting == ( UBaseType_t ) 0 )
```

**NULL Pointer Dereference\Path 19:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1300 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c in line 2936 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c in line 2209.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c |
| Line | 2939 | 2225 |
| Object | null | u |

Code Snippet

|  |  |
|---|---|
| File Name | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c |
| Method | QueueSetMemberHandle_t xQueueSelectFromSet( QueueSetHandle_t xQueueSet, |

```
....
2939.           QueueSetMemberHandle_t xReturn = NULL;
```

▼

|  |  |
|---|---|
| File Name | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c |
| Method | static void prvCopyDataFromQueue( Queue_t * const pxQueue, |

```
....
2225.           ( void ) memcpy( ( void * ) pvBuffer, ( void * ) pxQueue->u.xQueue.pcReadFrom, ( size_t ) pxQueue->uxItemSize ); /*lint !e961 !e418 !e9087 MISRA exception as the casts are only redundant for some ports.  Also previous logic ensures a null pointer can only be passed to memcpy() when the count is 0.  Cast to void required by function signature and safe as no alignment requirement and copy length specified in bytes. */
```

### NULL Pointer Dereference\Path 20:

|  |  |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1301 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c in line 2950 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c in line 2209.

|  | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c |
| Line | 2952 | 2225 |
| Object | null | u |

Code Snippet

|  |  |
|---|---|
| File Name | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c |
| Method | QueueSetMemberHandle_t xQueueSelectFromSetFromISR( QueueSetHandle_t xQueueSet ) |

```
....
2952.           QueueSetMemberHandle_t xReturn = NULL;
```

▼

|  |  |
|---|---|
| File Name | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c |
| Method | static void prvCopyDataFromQueue( Queue_t * const pxQueue, |

```
....
2225.           ( void ) memcpy( ( void * ) pvBuffer, ( void * ) pxQueue->u.xQueue.pcReadFrom, ( size_t ) pxQueue->uxItemSize ); /*lint !e961 !e418 !e9087 MISRA exception as the casts are only redundant for some ports.  Also previous logic ensures a null pointer can only be passed to memcpy() when the count is 0.  Cast to void required by function signature and safe as no alignment requirement and copy length specified in bytes. */
```

**NULL Pointer Dereference\Path 21:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1302 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c in line 2936 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c in line 2209.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c |
| Line | 2939 | 2216 |
| Object | null | u |

Code Snippet

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c

Method QueueSetMemberHandle_t xQueueSelectFromSet( QueueSetHandle_t xQueueSet,

```
....
2939.           QueueSetMemberHandle_t xReturn = NULL;
```

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c

Method static void prvCopyDataFromQueue( Queue_t * const pxQueue,

```
....
2216.           if( pxQueue->u.xQueue.pcReadFrom >= pxQueue->u.xQueue.pcTail ) /*lint !e946 MISRA exception justified as use of the relational operator is the cleanest solutions. */
```

**NULL Pointer Dereference\Path 22:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1303 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c in line 2950 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c in line 2209.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c |
| Line | 2952 | 2216 |
| Object | null | u |

Code Snippet
File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c
Method QueueSetMemberHandle_t xQueueSelectFromSetFromISR( QueueSetHandle_t xQueueSet )

```
....
2952.           QueueSetMemberHandle_t xReturn = NULL;
```

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c
Method static void prvCopyDataFromQueue( Queue_t * const pxQueue,

```
....
2216.           if( pxQueue->u.xQueue.pcReadFrom >= pxQueue->u.xQueue.pcTail ) /*lint !e946 MISRA exception justified as use of the relational operator is the cleanest solutions. */
```

**NULL Pointer Dereference\Path 23:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1304 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c in line 2936 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c in line 2209.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c |
| Line | 2939 | 2216 |
| Object | null | u |

Code Snippet
File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c
Method QueueSetMemberHandle_t xQueueSelectFromSet( QueueSetHandle_t xQueueSet,

```
....
2939.          QueueSetMemberHandle_t xReturn = NULL;
```

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c

Method static void prvCopyDataFromQueue( Queue_t * const pxQueue,

```
....
2216.          if( pxQueue->u.xQueue.pcReadFrom >= pxQueue-
>u.xQueue.pcTail ) /*lint !e946 MISRA exception justified as use of the
relational operator is the cleanest solutions. */
```

### NULL Pointer Dereference\Path 24:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1305 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c in line 2950 is not initialized when it is used by u at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c in line 2209.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c |
| Line | 2952 | 2216 |
| Object | null | u |

Code Snippet

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c

Method QueueSetMemberHandle_t xQueueSelectFromSetFromISR( QueueSetHandle_t xQueueSet )

```
....
2952.          QueueSetMemberHandle_t xReturn = NULL;
```

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c

Method static void prvCopyDataFromQueue( Queue_t * const pxQueue,

```
....
2216.          if( pxQueue->u.xQueue.pcReadFrom >= pxQueue-
>u.xQueue.pcTail ) /*lint !e946 MISRA exception justified as use of the
relational operator is the cleanest solutions. */
```

### NULL Pointer Dereference\Path 25:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1306 |
| Status | New |

The variable declared in null at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c in line 2936 is not initialized when it is used by pxQueue at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c in line 2350.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c |
| Line | 2939 | 2356 |
| Object | null | pxQueue |

Code Snippet

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c

Method QueueSetMemberHandle_t xQueueSelectFromSet( QueueSetHandle_t xQueueSet,

```
....
2939.           QueueSetMemberHandle_t xReturn = NULL;
```

File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-31571-TP.c

Method static BaseType_t prvIsQueueEmpty( const Queue_t * pxQueue )

```
....
2356.           if( pxQueue->uxMessagesWaiting == ( UBaseType_t ) 0 )
```

**NULL Pointer Dereference\Path 26:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1307 |
| Status | New |

The variable declared in 0 at FreeRTOS@@FreeRTOS-Kernel-V10.3.0-CVE-2023-36328-TP.c in line 1247 is not initialized when it is used by a at FreeRTOS@@FreeRTOS-Kernel-V10.3.0-CVE-2023-36328-TP.c in line 1247.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.3.0-CVE-2023-36328-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.3.0-CVE-2023-36328-TP.c |
| Line | 1251 | 1251 |
| Object | 0 | a |

Code Snippet
File Name FreeRTOS@@FreeRTOS-Kernel-V10.3.0-CVE-2023-36328-TP.c
Method void mp_set (mp_int * a, mp_digit b)

```
....
1251.     a->used  = (a->dp[0] != 0) ? 1 : 0;
```

**NULL Pointer Dereference\Path 27:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1308 |
| Status | New |

The variable declared in 0 at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-32020-TP.c in line 132 is not initialized when it is used by xStart at FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-32020-TP.c in line 132.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-32020-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-32020-TP.c |
| Line | 154 | 154 |
| Object | 0 | xStart |

Code Snippet
File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.0-kernel-only-CVE-2021-32020-TP.c
Method static void prvHeapInit( void )

```
....
154.      xStart.xBlockSize = ( size_t ) 0;
```

**NULL Pointer Dereference\Path 28:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1309 |
| Status | New |

The variable declared in 0 at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-CVE-2021-32020-FP.c in line 132 is not initialized when it is used by xStart at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-CVE-2021-32020-FP.c in line 132.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-CVE-2021-32020-FP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-CVE-2021-32020-FP.c |
| Line | 154 | 154 |
| Object | 0 | xStart |

|  |  |
|---|---|
| Code Snippet |  |
| File Name | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-CVE-2021-32020-FP.c |
| Method | static void prvHeapInit( void ) |

```
....
154.        xStart.xBlockSize = ( size_t ) 0;
```

**NULL Pointer Dereference\Path 29:**

|  |  |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1310 |
| Status | New |

The variable declared in 0 at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-32020-TP.c in line 134 is not initialized when it is used by xStart at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-32020-TP.c in line 134.

|  | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-32020-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-32020-TP.c |
| Line | 156 | 156 |
| Object | 0 | xStart |

|  |  |
|---|---|
| Code Snippet |  |
| File Name | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-1-CVE-2021-32020-TP.c |
| Method | static void prvHeapInit( void ) |

```
....
156.        xStart.xBlockSize = ( size_t ) 0;
```

**NULL Pointer Dereference\Path 30:**

|  |  |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1311 |
| Status | New |

The variable declared in 0 at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-32020-TP.c in line 134 is not initialized when it is used by xStart at FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-32020-TP.c in line 134.

|  | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-32020-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-32020-TP.c |
| Line | 156 | 156 |

| | | |
|---|---|---|
| Object | 0 | xStart |

Code Snippet
File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.3-LTS-Patch-3-CVE-2021-32020-TP.c
Method static void prvHeapInit( void )

```
....
156.        xStart.xBlockSize = ( size_t ) 0;
```

**NULL Pointer Dereference\Path 31:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1312 |
| Status | New |

The variable declared in 0 at FreeRTOS@@FreeRTOS-Kernel-V10.4.4-CVE-2021-32020-FP.c in line 133 is not initialized when it is used by xStart at FreeRTOS@@FreeRTOS-Kernel-V10.4.4-CVE-2021-32020-FP.c in line 133.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.4.4-CVE-2021-32020-FP.c | FreeRTOS@@FreeRTOS-Kernel-V10.4.4-CVE-2021-32020-FP.c |
| Line | 155 | 155 |
| Object | 0 | xStart |

Code Snippet
File Name FreeRTOS@@FreeRTOS-Kernel-V10.4.4-CVE-2021-32020-FP.c
Method static void prvHeapInit( void )

```
....
155.        xStart.xBlockSize = ( size_t ) 0;
```

**NULL Pointer Dereference\Path 32:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1313 |
| Status | New |

The variable declared in 0 at FreeRTOS@@FreeRTOS-Kernel-V10.6.0-CVE-2021-32020-TP.c in line 136 is not initialized when it is used by xStart at FreeRTOS@@FreeRTOS-Kernel-V10.6.0-CVE-2021-32020-TP.c in line 136.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.6.0-CVE-2021-32020-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.6.0-CVE-2021-32020-TP.c |

| | | |
|---|---|---|
| Line | 158 | 158 |
| Object | 0 | xStart |

Code Snippet
File Name FreeRTOS@@FreeRTOS-Kernel-V10.6.0-CVE-2021-32020-TP.c
Method static void prvHeapInit( void )

```
....
158.        xStart.xBlockSize = ( size_t ) 0;
```

**NULL Pointer Dereference\Path 33:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1314 |
| Status | New |

The variable declared in 0 at FreeRTOS@@FreeRTOS-Kernel-V10.6.2-CVE-2021-32020-TP.c in line 136 is not initialized when it is used by xStart at FreeRTOS@@FreeRTOS-Kernel-V10.6.2-CVE-2021-32020-TP.c in line 136.

| | Source | Destination |
|---|---|---|
| File | FreeRTOS@@FreeRTOS-Kernel-V10.6.2-CVE-2021-32020-TP.c | FreeRTOS@@FreeRTOS-Kernel-V10.6.2-CVE-2021-32020-TP.c |
| Line | 158 | 158 |
| Object | 0 | xStart |

Code Snippet
File Name FreeRTOS@@FreeRTOS-Kernel-V10.6.2-CVE-2021-32020-TP.c
Method static void prvHeapInit( void )

```
....
158.        xStart.xBlockSize = ( size_t ) 0;
```

## 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=1000019&projectid=14&pathid=1271 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2020- | FreeRDP@@FreeRDP-2.0.0-CVE-2020- |

| | | |
|---|---|---|
| | 13398-TP.c | 13398-TP.c |
| Line | 481 | 481 |
| Object | sizeof | sizeof |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2020-13398-TP.c
Method static void string_list_allocate(string_list* list, int allocate_count)

```
....
481.          list->strings = calloc((size_t)allocate_count,
sizeof(char*));
```

**Use of Sizeof On a Pointer Type\Path 2:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1272 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1708 | 1708 |
| Object | sizeof | sizeof |

Code Snippet
File Name FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c
Method static BOOL wf_cliprdr_array_ensure_capacity(wfClipboard* clipboard)

```
....
1708.                                          new_size *
sizeof(FILEDESCRIPTORW*));
```

**Use of Sizeof On a Pointer Type\Path 3:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1273 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1713 | 1713 |
| Object | sizeof | sizeof |

| | |
|---|---|
| Code Snippet | |
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Method | static BOOL wf_cliprdr_array_ensure_capacity(wfClipboard* clipboard) |

```
....
1713.               new_name = (WCHAR**)realloc(clipboard->file_names,
new_size * sizeof(WCHAR*));
```

### Use of Sizeof On a Pointer Type\Path 4:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1274 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1709 | 1709 |
| Object | sizeof | sizeof |

| | |
|---|---|
| Code Snippet | |
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | static BOOL wf_cliprdr_array_ensure_capacity(wfClipboard* clipboard) |

```
....
1709.                                                  new_size *
sizeof(FILEDESCRIPTORW*));
```

### Use of Sizeof On a Pointer Type\Path 5:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1275 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1714 | 1714 |
| Object | sizeof | sizeof |

| | |
|---|---|
| Code Snippet | |
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | static BOOL wf_cliprdr_array_ensure_capacity(wfClipboard* clipboard) |

```
....
1714.              new_name = (WCHAR**)realloc(clipboard->file_names,
new_size * sizeof(WCHAR*));
```

### Use of Sizeof On a Pointer Type\Path 6:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1276](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1276) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 1709 | 1709 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Method | static BOOL wf_cliprdr_array_ensure_capacity(wfClipboard* clipboard) |

```
....
1709.                                                         new_size *
sizeof(FILEDESCRIPTORW*));
```

### Use of Sizeof On a Pointer Type\Path 7:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | [http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1277](http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1277) |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 1714 | 1714 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Method | static BOOL wf_cliprdr_array_ensure_capacity(wfClipboard* clipboard) |

```
....
1714.              new_name = (WCHAR**)realloc(clipboard->file_names,
new_size * sizeof(WCHAR*));
```

### Use of Sizeof On a Pointer Type\Path 8:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1278 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Line | 810 | 810 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-beta1-CVE-2024-32662-TP.c |
| Method | static state_run_t rdp_recv_server_redirection_pdu(rdpRdp* rdp, wStream* s) |

```
....
810.                 redirection->TargetNetAddresses =
(char**)calloc(count, sizeof(char*));
```

### Use of Sizeof On a Pointer Type\Path 9:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1279 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Line | 833 | 833 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.0.0-rc0-CVE-2024-32662-TP.c |
| Method | static state_run_t rdp_recv_server_redirection_pdu(rdpRdp* rdp, wStream* s) |

```
....
833.                        redirection->TargetNetAddresses =
(char**)calloc(count, sizeof(char*));
```

### Use of Sizeof On a Pointer Type\Path 10:

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

| | |
|---|---|
| | PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1280 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Line | 127 | 127 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Method | static BOOL redirection_copy_array(char*** dst, UINT32* plen, const char** str, size_t len) |

```
....
127.        *dst = calloc(len, sizeof(char*));
```

**Use of Sizeof On a Pointer Type\Path 11:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1281 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Line | 821 | 821 |
| Object | sizeof | sizeof |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-3.4.0-CVE-2024-32662-TP.c |
| Method | static state_run_t rdp_recv_server_redirection_pdu(rdpRdp* rdp, wStream* s) |

```
....
821.                    redirection->TargetNetAddresses = 
(char**)calloc(count, sizeof(char*));
```

## 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=1000019&projectid=14&pathid=1183 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1634 | 1634 |
| Object | CreateFileW | CreateFileW |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Method | static BOOL wf_cliprdr_get_file_contents(WCHAR* file_name, BYTE* buffer, LONG positionLow, |

```
....
1634.        hFile = CreateFileW(file_name, GENERIC_READ,
FILE_SHARE_READ, NULL, OPEN_EXISTING,
```

### Incorrect Permission Assignment For Critical Resources\Path 2:

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1184 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Line | 1673 | 1673 |
| Object | CreateFileW | CreateFileW |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.0.0-CVE-2021-37595-TP.c |
| Method | static FILEDESCRIPTORW* wf_cliprdr_get_file_descriptor(WCHAR* file_name, size_t pathLen) |

```
....
1673.        hFile = CreateFileW(file_name, GENERIC_READ,
FILE_SHARE_READ, NULL, OPEN_EXISTING,
```

### Incorrect Permission Assignment For Critical Resources\Path 3:

| | |
|---|---|
| Severity | Low |

| | |
|---|---|
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1185 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1635 | 1635 |
| Object | CreateFileW | CreateFileW |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | static BOOL wf_cliprdr_get_file_contents(WCHAR* file_name, BYTE* buffer, LONG positionLow, |

```
....
1635.       hFile = CreateFileW(file_name, GENERIC_READ,
FILE_SHARE_READ, NULL, OPEN_EXISTING,
```

**Incorrect Permission Assignment For Critical Resources\Path 4:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1186 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Line | 1674 | 1674 |
| Object | CreateFileW | CreateFileW |

Code Snippet

| | |
|---|---|
| File Name | FreeRDP@@FreeRDP-2.2.0-CVE-2021-37595-TP.c |
| Method | static FILEDESCRIPTORW* wf_cliprdr_get_file_descriptor(WCHAR* file_name, size_t pathLen) |

```
....
1674.       hFile = CreateFileW(file_name, GENERIC_READ,
FILE_SHARE_READ, NULL, OPEN_EXISTING,
```

**Incorrect Permission Assignment For Critical Resources\Path 5:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14 |

| | |
|---|---|
| | &pathid=1187 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 1635 | 1635 |
| Object | CreateFileW | CreateFileW |

Code Snippet
File Name FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c
Method static BOOL wf_cliprdr_get_file_contents(WCHAR* file_name, BYTE* buffer, LONG positionLow,

```
....
1635.       hFile = CreateFileW(file_name, GENERIC_READ,
FILE_SHARE_READ, NULL, OPEN_EXISTING,
```

**Incorrect Permission Assignment For Critical Resources\Path 6:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1188 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c | FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c |
| Line | 1674 | 1674 |
| Object | CreateFileW | CreateFileW |

Code Snippet
File Name FreeRDP@@FreeRDP-2.3.0-CVE-2021-37595-TP.c
Method static FILEDESCRIPTORW* wf_cliprdr_get_file_descriptor(WCHAR* file_name, size_t pathLen)

```
....
1674.       hFile = CreateFileW(file_name, GENERIC_READ,
FILE_SHARE_READ, NULL, OPEN_EXISTING,
```

## Information Exposure Through Comments

Query Path:
CPP\Cx\CPP Low Visibility\Information Exposure Through Comments Version:1

### Categories

FISMA 2014: Identification And Authentication
NIST SP 800-53: SC-28 Protection of Information at Rest (P1)

Description

**Information Exposure Through Comments\Path 1:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1189 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c |
| Line | 496 | 496 |
| Object | password 'p | password 'p |

Code Snippet

File Name freeswitch@@sofia-sip-v1.13.10-CVE-2023-22741-TP.c

Method * STUN password 'pwd'. The received content should be

```
....
496.     * STUN password 'pwd'. The received content should be
```

**Information Exposure Through Comments\Path 2:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1190 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c |
| Line | 496 | 496 |
| Object | password 'p | password 'p |

Code Snippet

File Name freeswitch@@sofia-sip-v1.13.2-CVE-2023-22741-TP.c

Method * STUN password 'pwd'. The received content should be

```
....
496.     * STUN password 'pwd'. The received content should be
```

**Information Exposure Through Comments\Path 3:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1191 |

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

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c |
| Line | 496 | 496 |
| Object | password 'p | password 'p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.3-CVE-2023-22741-TP.c
Method * STUN password 'pwd'. The received content should be

```
....
496.    * STUN password 'pwd'. The received content should be
```

**Information Exposure Through Comments\Path 4:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1192 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c |
| Line | 496 | 496 |
| Object | password 'p | password 'p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.4-CVE-2023-22741-TP.c
Method * STUN password 'pwd'. The received content should be

```
....
496.    * STUN password 'pwd'. The received content should be
```

**Information Exposure Through Comments\Path 5:**

| | |
|---|---|
| Severity | Low |
| Result State | To Verify |
| Online Results | http://WIN-PTJMSNK3USL/CxWebClient/ViewerMain.aspx?scanid=1000019&projectid=14&pathid=1193 |
| Status | New |

| | Source | Destination |
|---|---|---|
| File | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c | freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c |

| | | |
|---|---|---|
| Line | 496 | 496 |
| Object | password 'p | password 'p |

Code Snippet
File Name freeswitch@@sofia-sip-v1.13.6-CVE-2023-22741-TP.c
Method * STUN password 'pwd'. The received content should be

```
....
496.    * STUN password 'pwd'. The received content should be
```

# 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

# Divide By Zero

## Risk

### What might happen

When a program divides a number by zero, an exception will be raised. If this exception is not handled by the application, unexpected results may occur, including crashing the application. This can be considered a DoS (Denial of Service) attack, if an external user has control of the value of the denominator or can cause this error to occur.

## Cause

### How does it happen

The program receives an unexpected value, and uses it for division without filtering, validation, or verifying that the value is not zero. The application does not explicitly handle this error or prevent division by zero from occuring.

## General Recommendations

### How to avoid it

- Before dividing by an unknown value, validate the number and explicitly ensure it does not evaluate to zero.
- Validate all untrusted input from all sources, in particular verifying that it is not zero before dividing with it.
- Verify output of methods, calculations, dictionary lookups, and so on, and ensure it is not zero before dividing with the result.
- Ensure divide-by-zero errors are caught and handled appropriately.

## Source Code Examples

### Java

### Divide by Zero

```java
public float getAverage(HttpServletRequest req) {
      int total = Integer.parseInt(req.getParameter("total"));
      int count = Integer.parseInt(req.getParameter("count"));

      return total / count;
}
```

### Checked Division

```java
public float getAverage(HttpServletRequest req) {
      int total = Integer.parseInt(req.getParameter("total"));
      int count = Integer.parseInt(req.getParameter("count"));
```

```
        if (count > 0)
                return total / count;
        else
                return 0;
}
```

# Buffer Overflow boundcpy WrongSizeParam

## Risk

### What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

## Cause

### How does it happen

Buffer Overflows can manifest in numerous different variations. In it's most basic form, the attack controls a buffer, which is then copied to a smaller buffer without size verification. Because the attacker's source buffer is larger than the program's target buffer, the attacker's data overwrites whatever is next on the stack, allowing the attacker to control program structures.

Alternatively, the vulnerability could be the result of improper bounds checking; exposing internal memory addresses outside of their valid scope; allowing the attacker to control the size of the target buffer; or various other forms.

## General Recommendations

### How to avoid it

- Always perform proper bounds checking before copying buffers or strings.
- Prefer to use safer functions and structures, e.g. safe string classes over `char*`, `strncpy` over `strcpy`, and so on.
- Consistently apply tests for the size of buffers.
- Do not return variable addresses outside the scope of their variables.

## Source Code Examples

### CPP

### Overflowing Buffers

```
const int BUFFER_SIZE = 10;
char buffer[BUFFER_SIZE];

void copyStringToBuffer(char* inputString)
{
    strcpy(buffer, inputString);
}
```

### Checked Buffers

```
const int BUFFER_SIZE = 10;
const int MAX_INPUT_SIZE = 256;
```

```
char buffer[BUFFER_SIZE];

void copyStringToBuffer(char* inputString)
{
    if (strnlen(inputString, MAX_INPUT_SIZE) < sizeof(buffer))
    {
        strncpy(buffer, inputString, sizeof(buffer));
    }
}
```

# MemoryFree on StackVariable

## Risk

**What might happen**

Undefined Behavior may result with a crash. Crashes may give an attacker valuable information about the system and the program internals. Furthermore, it may leave unprotected files (e.g memory) that may be exploited.

## Cause

**How does it happen**

Calling free() on a variable that was not dynamically allocated (e.g. malloc) will result with an Undefined Behavior.

## General Recommendations

**How to avoid it**

Use free() only on dynamically allocated variables in order to prevent unexpected behavior from the compiler.

## Source Code Examples

### CPP

**Bad - Calling free() on a static variable**

```
void clean_up(){
  char temp[256];
  do_something();
  free(tmp);
  return;
}
```

**Good - Calling free() only on variables that were dynamically allocated**

```
void clean_up(){
  char *buff;
  buff = (char*) malloc(1024);
  free(buff);
  return;
}
```

# Wrong Size t Allocation

## Risk

### What might happen

Incorrect allocation of memory may result in unexpected behavior by either overwriting sections of memory with unexpected values. Under certain conditions where both an incorrect allocation of memory and the values being written can be controlled by an attacker, such an issue may result in execution of malicious code.

## Cause

### How does it happen

Some memory allocation functions require a size value to be provided as a parameter. The allocated size should be derived from the provided value, by providing the length value of the intended source, multiplied by the size of that length. Failure to perform the correct arithmetic to obtain the exact size of the value will likely result in the source overflowing its destination.

## General Recommendations

### How to avoid it

- Always perform the correct arithmetic to determine size.
- Specifically for memory allocation, calculate the allocation size from the allocation source:
    - Derive the size value from the length of intended source to determine the amount of units to be processed.
    - Always programmatically consider the size of the each unit and their conversion to memory units - for example, by using sizeof() on the unit's type.
    - Memory allocation should be a multiplication of the amount of units being written, times the size of each unit.

## Source Code Examples

### CPP

### Allocating and Assigning Memory without Sizeof Arithmetic

```cpp
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

```cpp
int *ptr;
ptr = (int*)malloc(5 * sizeof(int));
```

```
for (int i = 0; i < 5; i++)
{
      ptr[i] = i * 2 + 1;
}
```

### Incorrect Arithmetic of Multi-Byte String Allocation

```
wchar_t * dest;
dest = (wchar_t *)malloc(wcslen(source) + 1); // Would not crash for a short "source"
wcscpy((wchar_t *)dest, source);
wprintf(L"Dest: %s\r\n", dest);
```

### Correct Arithmetic of Multi-Byte String Allocation

```
wchar_t * dest;
dest = (wchar_t *)malloc((wcslen(source) + 1) * sizeof(wchar_t));
wcscpy((wchar_t *)dest, source);
wprintf(L"Dest: %s\r\n", dest);
```

# Char Overflow

## Risk

### What might happen

Assigning large data types into smaller data types, without proper checks and explicit casting, will lead to undefined behavior and unintentional effects, such as data corruption (e.g. value wraparound, wherein maximum values become minimum values); system crashes; infinite loops; logic errors, such as bypassing of security mechanisms; or even buffer overflows leading to arbitrary code execution.

## Cause

### How does it happen

This flaw can occur when implicitly casting numerical data types of a larger size, into a variable with a data type of a smaller size. This forces the program to discard some bits of information from the number. Depending on how the numerical data types are stored in memory, this is often the bits with the highest value, causing substantial corruption of the stored number. Alternatively, the sign bit of a signed integer could be lost, completely reversing the intention of the number.

## General Recommendations

### How to avoid it

- Avoid casting larger data types to smaller types.
- Prefer promoting the target variable to a large enough data type.
- If downcasting is necessary, always check that values are valid and in range of the target type, before casting

## Source Code Examples

### CPP

### Unsafe Downsize Casting

```
int unsafe_addition(short op1, int op2) {

    // op2 gets forced from int into a short
    short total = op1 + op2;

    return total;
}
```

### Safer Use of Proper Data Types

```
int safe_addition(short op1, int op2) {

    // total variable is of type int, the largest type that is needed
    int total = 0;

    // check if total will overflow available integer size
    if (INT_MAX - abs(op2) > op1)
    {
        total = op1 + op2;
    }
    else
    {
        // instead of overflow, saturate (but this is not always a good thing)
        total = INT_MAX
    }

    return total;
}
```

# Integer Overflow

## Risk

### What might happen

Assigning large data types into smaller data types, without proper checks and explicit casting, will lead to undefined behavior and unintentional effects, such as data corruption (e.g. value wraparound, wherein maximum values become minimum values); system crashes; infinite loops; logic errors, such as bypassing of security mechanisms; or even buffer overflows leading to arbitrary code execution.

## Cause

### How does it happen

This flaw can occur when implicitly casting numerical data types of a larger size, into a variable with a data type of a smaller size. This forces the program to discard some bits of information from the number. Depending on how the numerical data types are stored in memory, this is often the bits with the highest value, causing substantial corruption of the stored number. Alternatively, the sign bit of a signed integer could be lost, completely reversing the intention of the number.

## General Recommendations

### How to avoid it

- Avoid casting larger data types to smaller types.
- Prefer promoting the target variable to a large enough data type.
- If downcasting is necessary, always check that values are valid and in range of the target type, before casting

## Source Code Examples

# Dangerous Functions

## Risk

### What might happen

Use of dangerous functions may expose varying risks associated with each particular function, with potential impact of improper usage of these functions varying significantly. The presence of such functions indicates a flaw in code maintenance policies and adherence to secure coding practices, in a way that has allowed introducing known dangerous code into the application.

## Cause

### How does it happen

A dangerous function has been identified within the code. Functions are often deemed dangerous to use for numerous reasons, as there are different sets of vulnerabilities associated with usage of such functions. For example, some string copy and concatenation functions are vulnerable to Buffer Overflow, Memory Disclosure, Denial of Service and more. Use of these functions is not recommended.

## General Recommendations

### How to avoid it

- Deploy a secure and recommended alternative to any functions that were identified as dangerous.
    - If no secure alternative is found, conduct further researching and testing to identify whether current usage successfully sanitizes and verifies values, and thus successfully avoids the use-cases for whom the function is indeed dangerous
- Conduct a periodical review of methods that are in use, to ensure that all external libraries and built-in functions are up-to-date and whose use has not been excluded from best secure coding practices.

## Source Code Examples

### CPP

### Buffer Overflow in gets()

```
int main()
{
      char buf[10];
      printf("Please enter your name: ");
      gets(buf); // veryveryverylongname
      if (buf == ACCEPTED_NAME)
      {
            // Do something
      }
      return 0;
}
```

### Safe reading from user

```
int main()
{
      char buf[10];

      printf("Please enter your name: ");
      fgets(buf, sizeof(buf), stdin); //setting the amount of bytes to read
      if (buf == ACCEPTED_NAME)
      {
             //Do something
      }
      return 0;
}
```

### Unsafe function for string copy

```
int main(int argc, char* argv[])
{
      char buf[10];
      strcpy(buf, argv[1]); // overflow occurs when len(argv[1]) > 10 bytes

      return 0;
}
```

### Safe string copy

```
int main(int argc, char* argv[])
{
      char buf[10];
      strncpy(buf, argv[1], sizeof(buf));
      buf[9]= '\0'; //strncpy doesn't NULL terminates

      return 0;
}
```

### Unsafe format string

```
int main(int argc, char* argv[])
{
      printf(argv[1]); // If argv[1] contains a format token, such as %s,%x or %d, will cause
an access violation
      return 0;
}
```

### Safe format string

```
int main(int argc, char* argv[])
{
      printf("%s", argv[1]); // Second parameter is not a formattable string
      return 0;
}
```

# Heap Inspection

## Risk

### What might happen

All variables stored by the application in unencrypted memory can potentially be retrieved by an unauthorized user, with privlieged access to the machine. For example, a privileged attacker could attach a debugger to the running process, or retrieve the process's memory from the swapfile or crash dump file.
Once the attacker finds the user passwords in memory, these can be reused to easily impersonate the user to the system.

## Cause

### How does it happen

String variables are immutable - in other words, once a string variable is assigned, its value cannot be changed or removed. Thus, these strings may remain around in memory, possibly in multiple locations, for an indefinite period of time until the garbage collector happens to remove it. Sensitive data, such as passwords, will remain exposed in memory as plaintext with no control over their lifetime.

## General Recommendations

### How to avoid it

Generic Guidance:

- Do not store senstiive data, such as passwords or encryption keys, in memory in plaintext, even for a short period of time.
- Prefer to use specialized classes that store encrypted memory.
- Alternatively, store secrets temporarily in mutable data types, such as byte arrays, and then promptly zeroize the memory locations.

Specific Recommendations - Java:

- Instead of storing passwords in immutable strings, prefer to use an encrypted memory object, such as SealedObject.

Specific Recommendations - .NET:

- Instead of storing passwords in immutable strings, prefer to use an encrypted memory object, such as SecureString or ProtectedData.

## Source Code Examples

### Java

**Plaintext Password in Immutable String**

```
class Heap_Inspection
{
  private string password;
  void setPassword()
 {
      password = System.console().readLine("Enter your password: ");
  }
}
```

### Password Protected in Memory

```
class Heap_Inspection_Fixed
{
  private SealedObject password;

  void setPassword()
 {
      byte[] sKey = getKeyFromConfig();
      Cipher c = Cipher.getInstance("AES");
      c.init(Cipher.ENCRYPT_MODE, sKey);

      char[] input = System.console().readPassword("Enter your password: ");
      password = new SealedObject(Arrays.asList(input), c);

    //Zero out the possible password, for security.
    Arrays.fill(password, '0');
  }
}
```

### CPP

### Vulnerable C code

```
/* Vulnerable to heap inspection */

#include <stdio.h>


void somefunc(){
      printf("Yea, I'm just being called for the heap of it..\n");
}

void authfunc(){
        char* password = (char *) malloc(256);
        char ch;
        ssize_t k;
            int i=0;
        while(k = read(0, &ch, 1) > 0)
        {
                if (ch == '\n'){
                        password[i]='\0';
                        break;
                } else{
                        password[i++]=ch;
                        fflush(0);
                }
        }
        printf("Password: %s\n",&password[0]);
}
```

```
int main()

{

    printf("Please enter a password:\n");

      authfunc();
      printf("You can now dump memory to find this password!");
      somefunc();
      gets();

}
```

## Safe C code

```
/* Pesumably safe heap */

#include <stdio.h>
#include <string.h>

#define STDIN_FILENO 0

void somefunc(){
        printf("Yea, I'm just being called for the heap of it..\n");
}

void authfunc(){
      char* password = (char*) malloc(256);
      int i=0;
      char ch;
      ssize_t k;
      while(k = read(STDIN_FILENO, &ch, 1) > 0)
      {
            if (ch == '\n'){
                   password[i]='\0';
                   break;
            } else{
                   password[i++]=ch;
                   fflush(0);
            }
      }
      i=0;
      memset(password,'\0',256);
}

int main()

{

      printf("Please enter a password:\n");
      authfunc();
      somefunc();
      char ch;
      while(read(STDIN_FILENO, &ch, 1) > 0)
      {
            if (ch == '\n')
                   break;
      }
}
```

## Failure to Release Memory Before Removing Last Reference ('Memory Leak')

**Weakness ID:** 401 *(Weakness Base)* **Status:** Draft

### Description

#### Description Summary

The software does not sufficiently track and release allocated memory after it has been used, which slowly consumes remaining memory.

#### Extended Description

This is often triggered by improper handling of malformed data or unexpectedly interrupted sessions.

### Terminology Notes

"memory leak" has sometimes been used to describe other kinds of issues, e.g. for information leaks in which the contents of memory are inadvertently leaked (CVE-2003-0400 is one such example of this terminology conflict).

### Time of Introduction

- Architecture and Design
- Implementation

### Applicable Platforms

#### Languages

C

C++

### Modes of Introduction

Memory leaks have two common and sometimes overlapping causes:

- Error conditions and other exceptional circumstances
- Confusion over which part of the program is responsible for freeing the memory

### Common Consequences

| Scope | Effect |
|---|---|
| Availability | Most memory leaks result in general software reliability problems, but if an attacker can intentionally trigger a memory leak, the attacker might be able to launch a denial of service attack (by crashing or hanging the program) or take advantage of other unexpected program behavior resulting from a low memory condition. |

### Likelihood of Exploit

Medium

### Demonstrative Examples

#### Example 1

The following C function leaks a block of allocated memory if the call to read() fails to return the expected number of bytes:

*(Bad Code)*

*Example Language:* **C**

```
char* getBlock(int fd) {
char* buf = (char*) malloc(BLOCK_SIZE);
if (!buf) {
return NULL;
}
if (read(fd, buf, BLOCK_SIZE) != BLOCK_SIZE) {

return NULL;
}
return buf;
}
```

## Example 2

Here the problem is that every time a connection is made, more memory is allocated. So if one just opened up more and more connections, eventually the machine would run out of memory.

*(Bad Code)*

*Example Language:* **C**

```
bar connection(){
foo = malloc(1024);
return foo;
}
endConnection(bar foo) {

free(foo);
}
int main() {

while(1) //thread 1
//On a connection
foo=connection(); //thread 2
//When the connection ends
endConnection(foo)
}
```

### Observed Examples

| Reference | Description |
|---|---|
| CVE-2005-3119 | Memory leak because function does not free() an element of a data structure. |
| CVE-2004-0427 | Memory leak when counter variable is not decremented. |
| CVE-2002-0574 | Memory leak when counter variable is not decremented. |
| CVE-2005-3181 | Kernel uses wrong function to release a data structure, preventing data from being properly tracked by other code. |
| CVE-2004-0222 | Memory leak via unknown manipulations as part of protocol test suite. |
| CVE-2001-0136 | Memory leak via a series of the same command. |

### Potential Mitigations

Pre-design: Use a language or compiler that performs automatic bounds checking.

**Phase: Architecture and Design**

Use an abstraction library to abstract away risky APIs. Not a complete solution.

Pre-design through Build: The Boehm-Demers-Weiser Garbage Collector or valgrind can be used to detect leaks in code. This is not a complete solution as it is not 100% effective.

### Relationships

| Nature | Type | ID | Name | View(s) this relationship pertains to |
|---|---|---|---|---|
| ChildOf | Weakness Class | 398 | Indicator of Poor Code Quality | **Seven Pernicious Kingdoms (primary)700** |
| ChildOf | Category | 399 | Resource Management Errors | **Development Concepts (primary)699** |
| ChildOf | Category | 633 | Weaknesses that Affect Memory | **Resource-specific Weaknesses (primary)631** |
| ChildOf | Category | 730 | OWASP Top Ten 2004 Category A9 - Denial of Service | **Weaknesses in OWASP Top Ten (2004) (primary)711** |
| ChildOf | Weakness Base | 772 | Missing Release of Resource after Effective | **Research Concepts (primary)1000** |

| | | | | |
|---|---|---|---|---|
| | | | Lifetime | |
| MemberOf | View | 630 | Weaknesses Examined by SAMATE | **Weaknesses Examined by SAMATE (primary)630** |
| CanFollow | Weakness Class | 390 | Detection of Error Condition Without Action | Research Concepts1000 |

## Relationship Notes

This is often a resultant weakness due to improper handling of malformed data or early termination of sessions.

## Affected Resources

- Memory

## Functional Areas

- Memory management

## Taxonomy Mappings

| Mapped Taxonomy Name | Node ID | Fit | Mapped Node Name |
|---|---|---|---|
| PLOVER | | | Memory leak |
| 7 Pernicious Kingdoms | | | Memory Leak |
| CLASP | | | Failure to deallocate data |
| OWASP Top Ten 2004 | A9 | CWE More Specific | Denial of Service |

## White Box Definitions

A weakness where the code path has:

1. start statement that allocates dynamically allocated memory resource

2. end statement that loses identity of the dynamically allocated memory resource creating situation where dynamically allocated memory resource is never relinquished

Where "loses" is defined through the following scenarios:

1. identity of the dynamic allocated memory resource never obtained

2. the statement assigns another value to the data element that stored the identity of the dynamically allocated memory resource and there are no aliases of that data element

3. identity of the dynamic allocated memory resource obtained but never passed on to function for memory resource release

4. the data element that stored the identity of the dynamically allocated resource has reached the end of its scope at the statement and there are no aliases of that data element

## References

J. Whittaker and H. Thompson. "How to Break Software Security". Addison Wesley. 2003.

## Content History

**Submissions**

| Submission Date | Submitter | Organization | Source |
|---|---|---|---|
| | PLOVER | | Externally Mined |

**Modifications**

| Modification Date | Modifier | Organization | Source |
|---|---|---|---|
| 2008-07-01 | Eric Dalci | Cigital | External |
| | updated Time of Introduction | | |
| 2008-08-01 | | KDM Analytics | External |
| | added/updated white box definitions | | |
| 2008-08-15 | | Veracode | External |
| | Suggested OWASP Top Ten 2004 mapping | | |
| 2008-09-08 | CWE Content Team | MITRE | Internal |
| | updated Applicable Platforms, Common Consequences, Relationships, Other Notes, References, Relationship Notes, Taxonomy Mappings, Terminology Notes | | |
| 2008-10-14 | CWE Content Team | MITRE | Internal |
| | updated Description | | |
| 2009-03-10 | CWE Content Team | MITRE | Internal |
| | updated Other Notes | | |
| 2009-05-27 | CWE Content Team | MITRE | Internal |
| | updated Name | | |
| 2009-07-17 | KDM Analytics | | External |
| | Improved the White Box Definition | | |

|  |  |  |  |
|---|---|---|---|
| 2009-07-27 | CWE Content Team | MITRE | Internal |
|  | updated White Box Definitions |  |  |
| 2009-10-29 | CWE Content Team | MITRE | Internal |
|  | updated Modes of Introduction, Other Notes |  |  |
| 2010-02-16 | CWE Content Team | MITRE | Internal |
|  | updated Relationships |  |  |

### Previous Entry Names

| Change Date | Previous Entry Name |
|---|---|
| 2008-04-11 | Memory Leak |
| 2009-05-27 | Failure to Release Memory Before Removing Last Reference (aka 'Memory Leak') |

BACK TO TOP

# Use of a One Way Hash without a Salt

## Risk

### What might happen

If an attacker gains access to the hashed passwords, she would likely be able to reverse the hash due to this weakness, and retrieve the original password. Once the passwords are discovered, the attacker can impersonate the users, and take full advantage of their privileges and access their personal data. Furthermore, this would likely not be discovered, as the attacker is being identified solely by the victims' credentials.

## Cause

### How does it happen

Typical cryptographic hashes, such as SHA-1 and MD5, are incredibly fast. Combined with attack techniques such as precomputed Rainbow Tables, it is relatively easy for attackers to reverse the hashes, and discover the original passwords. Lack of a unique, random salt added to the password makes brute force attacks even simpler.

## General Recommendations

### How to avoid it

Generic Guidance:

- Always use strong, modern algorithms for encryption, hashing, and so on.

- Do not use weak, outdated, or obsolete algorithms.

- Ensure you select the correct cryptographic mechanism according to the specific requirements.

Specific Recommendations:

- Passwords should be protected using a password hashing algorithm, instead of a general cryptographic hash. This includes adaptive hashes such as bcrypt, scrypt, PBKDF2 and Argon2.

- Tune the work factor, or cost, of the adaptive hash function according to the designated environment and risk profile.

- Do not use a regular cryptographic hash, such as SHA-1 or MD5, to protect passwords, as these are too fast.

- If it is necessary to use a common hash to protect passwords, add several bytes of unique, random data ("salt") to the password before hashing it. Store the salt with the hashed password, and do not reuse the same salt for multiple passwords.

## Source Code Examples

### Java

### Unsalted Hashed Password

```
private String protectPassword(String password) {
        byte[] data = password.getBytes();
        byte[] hash = null;

        MessageDigest md = MessageDigest.getInstance("MD5");
        hash = md.digest(data);

        return Base64.getEncoder().encodeToString(hash);
}
```

### Fast Hash with Salt

```
private String protectPassword(String password) {
        byte[] data = password.getBytes("UTF-8");
        byte[] hash = null;

        try {
                MessageDigest md = MessageDigest.getInstance("SHA-1");

                SecureRandom rand = new SecureRandom();
                byte[] salt = new byte[32];
                rand.nextBytes(salt);

                md.update(salt);
                md.update(data);

                hash = md.digest();
      }
        catch (GeneralSecurityException gse) {
                handleCryptoErrors(gse);
      }
        finally {
                Arrays.fill(data, 0);
      }

        return Base64.getEncoder().encodeToString(hash);
}
```

### Slow, Adaptive Password Hash

```
private String protectPassword(String password) {
        byte[] data = password.getBytes("UTF-8");
        byte[] hash = null;

        try {
                SecureRandom rand = new SecureRandom();
                byte[] salt = new byte[32];
                rand.nextBytes(salt);

                SecretKeyFactory skf = SecretKeyFactory.getInstance("PBKDF2WithHmacSHA512");
                PBEKeySpec spec = new PBEKeySpec(data, salt, ITERATION_COUNT, KEY_LENGTH);
                // ITERATION_COUNT should be configured by environment, KEY_LENGTH should be 256
                SecretKey key = skf.generateSecret(spec);

                hash = key.getEncoded();
      }
        catch (GeneralSecurityException gse) {
                handleCryptoErrors(gse);
      }
        finally {
                Arrays.fill(data, 0);
      }

        return Base64.getEncoder().encodeToString(hash);
}
```

# 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());
```

## 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

## Information Leak Through Comments

**Weakness ID:** 615 *(Weakness Variant)* **Status:** Incomplete

### Description

## Description Summary

While adding general comments is very useful, some programmers tend to leave important data, such as: filenames related to the web application, old links or links which were not meant to be browsed by users, old code fragments, etc.

## Extended Description

An attacker who finds these comments can map the application's structure and files, expose hidden parts of the site, and study the fragments of code to reverse engineer the application, which may help develop further attacks against the site.

### Time of Introduction

- Implementation

### Demonstrative Examples

## Example 1

The following comment, embedded in a JSP, will be displayed in the resulting HTML output.

*(Bad Code)*

*Example Languages:* **HTML and JSP**

```
<!-- FIXME: calling this with more than 30 args kills the JDBC server -->
```

### Observed Examples

| Reference | Description |
|---|---|
| CVE-2007-6197 | Version numbers and internal hostnames leaked in HTML comments. |
| CVE-2007-4072 | CMS places full pathname of server in HTML comment. |
| CVE-2009-2431 | blog software leaks real username in HTML comment. |

### Potential Mitigations

Remove comments which have sensitive information about the design/implementation of the application. Some of the comments may be exposed to the user and affect the security posture of the application.

### Relationships

| Nature | Type | ID | Name | View(s) this relationship pertains to |
|---|---|---|---|---|
| ChildOf | Weakness Variant | 540 | Information Leak Through Source Code | **Development Concepts (primary)699**<br>**Research Concepts (primary)1000** |

### Content History

**Submissions**

| Submission Date | Submitter | Organization | Source |
|---|---|---|---|
| | Anonymous Tool Vendor (under NDA) | | Externally Mined |

**Modifications**

| Modification Date | Modifier | Organization | Source |
|---|---|---|---|
| 2008-07-01 | Sean Eidemiller | Cigital | External |
| | added/updated demonstrative examples | | |
| 2008-07-01 | Eric Dalci | Cigital | External |
| | updated Potential Mitigations, Time of Introduction | | |
| 2008-09-08 | CWE Content Team | MITRE | Internal |
| | updated Relationships, Taxonomy Mappings | | |
| 2008-10-14 | CWE Content Team | MITRE | Internal |
| | updated Description | | |
| 2009-03-10 | CWE Content Team | MITRE | Internal |

| | | | |
|---|---|---|---|
| | updated Demonstrative Examples | | |
| 2009-07-27 | CWE Content Team | MITRE | Internal |
| | updated Observed Examples, Taxonomy Mappings | | |

BACK TO TOP

# 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<br>updated Time of Introduction | Cigital | External |
| 2008-08-01 | added/updated white box definitions | KDM Analytics | External |
| 2008-09-08 | CWE Content Team<br>updated Applicable Platforms, Common Consequences, Relationships, Other Notes, Taxonomy Mappings, Weakness Ordinalities | MITRE | Internal |
| 2008-11-24 | CWE Content Team<br>updated Relationships, Taxonomy Mappings | MITRE | Internal |
| 2009-03-10 | CWE Content Team<br>updated Demonstrative Examples | MITRE | Internal |
| 2009-12-28 | CWE Content Team<br>updated Demonstrative Examples | MITRE | Internal |
| 2010-02-16 | CWE Content Team<br>updated Relationships | MITRE | Internal |

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 |