AliOS-Things-2 Scan Report

Project Name	AliOS-Things-2
Scan Start	Saturday, June 22, 2024 1:56:24 AM
Preset	Checkmarx Default
Scan Time	00h:03m:16s
Lines Of Code Scanned	35567
Files Scanned	20
Report Creation Time	Saturday, June 22, 2024 2:01:30 AM
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084
Team	CxServer
Checkmarx Version	8.7.0
Scan Type	Full
Source Origin	LocalPath
Density	10/1000 (Vulnerabilities/LOC)
Visibility	Public

Filter Settings

Severity

Included: High, Medium, Low, Information

Excluded: None

Result State

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

Excluded: None

Assigned to

Included: All

Categories

Included:

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

Excluded:

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

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

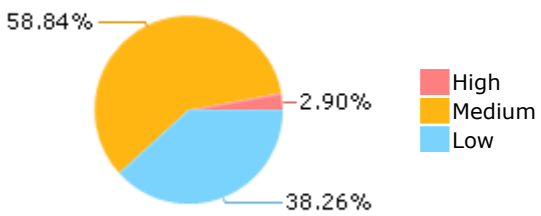
Results Limit

Results limit per query was set to 50

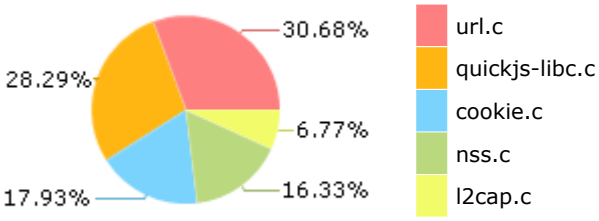
Selected Queries

Selected queries are listed in Result Summary

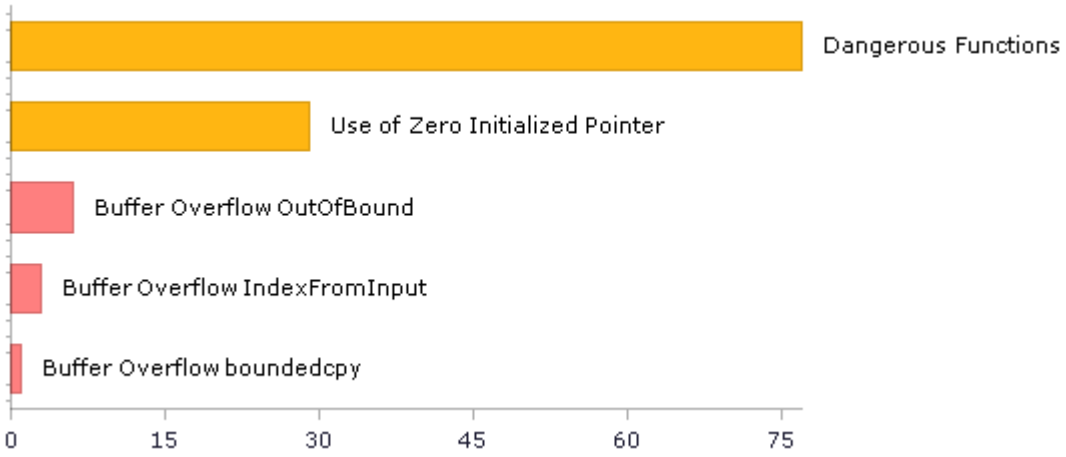
Result Summary



Most Vulnerable Files



Top 5 Vulnerabilities



Scan Summary - OWASP Top 10 2017

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

Category	Threat Agent	Exploitability	Weakness Prevalence	Weakness Detectability	Technical Impact	Business Impact	Issues Found	Best Fix Locations
A1-Injection	App. Specific	EASY	COMMON	EASY	SEVERE	App. Specific	59	31
A2-Broken Authentication	App. Specific	EASY	COMMON	AVERAGE	SEVERE	App. Specific	36	36
A3-Sensitive Data Exposure	App. Specific	AVERAGE	WIDESPREAD	AVERAGE	SEVERE	App. Specific	5	5
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	77	77
A10-Insufficient Logging & Monitoring	App. Specific	AVERAGE	WIDESPREAD	DIFFICULT	MODERATE	App. Specific	0	0

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

Scan Summary - OWASP Top 10 2013

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

Category	Threat Agent	Attack Vectors	Weakness Prevalence	Weakness Detectability	Technical Impact	Business Impact	Issues Found	Best Fix Locations
A1-Injection	EXTERNAL, INTERNAL, ADMIN USERS	EASY	COMMON	AVERAGE	SEVERE	ALL DATA	0	0
A2-Broken Authentication and Session Management	EXTERNAL, INTERNAL USERS	AVERAGE	WIDESPREAD	AVERAGE	SEVERE	AFFECTED DATA AND FUNCTIONS	0	0
A3-Cross-Site Scripting (XSS)	EXTERNAL, INTERNAL, ADMIN USERS	AVERAGE	VERY WIDESPREAD	EASY	MODERATE	AFFECTED DATA AND SYSTEM	0	0
A4-Insecure Direct Object References	SYSTEM USERS	EASY	COMMON	EASY	MODERATE	EXPOSED DATA	0	0
A5-Security Misconfiguration	EXTERNAL, INTERNAL, ADMIN USERS	EASY	COMMON	EASY	MODERATE	ALL DATA AND SYSTEM	0	0
A6-Sensitive Data Exposure	EXTERNAL, INTERNAL, ADMIN USERS, USERS BROWSERS	DIFFICULT	UNCOMMON	AVERAGE	SEVERE	EXPOSED DATA	2	2
A7-Missing Function Level Access Control*	EXTERNAL, INTERNAL USERS	EASY	COMMON	AVERAGE	MODERATE	EXPOSED DATA AND FUNCTIONS	0	0
A8-Cross-Site Request Forgery (CSRF)	USERS BROWSERS	AVERAGE	COMMON	EASY	MODERATE	AFFECTED DATA AND FUNCTIONS	0	0
A9-Using Components with Known Vulnerabilities*	EXTERNAL USERS, AUTOMATED TOOLS	AVERAGE	WIDESPREAD	DIFFICULT	MODERATE	AFFECTED DATA AND FUNCTIONS	77	77
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	32	29
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.	3	3
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.	31	31
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.	2	2
System And Communications Protection	Organizations must: (i) monitor, control, and protect organizational communications (i.e., information transmitted or received by organizational information systems) at the external boundaries and key internal boundaries of the information systems; and (ii) employ architectural designs, software development techniques, and systems engineering principles that promote effective information security within organizational information systems.	0	0
System And Information Integrity	Organizations must: (i) identify, report, and correct information and information system flaws in a timely manner; (ii) provide protection from malicious code at appropriate locations within organizational information systems; and (iii) monitor information system security alerts and advisories and take appropriate actions in response.	5	5

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

Scan Summary - NIST SP 800-53

Category	Issues Found	Best Fix Locations
AC-12 Session Termination (P2)	0	0
AC-3 Access Enforcement (P1)	37	37
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)	2	2
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)	1	1
SC-4 Information in Shared Resources (P1)	3	3
SC-5 Denial of Service Protection (P1)*	90	51
SC-8 Transmission Confidentiality and Integrity (P1)	0	0
SI-10 Information Input Validation (P1)*	37	34
SI-11 Error Handling (P2)*	16	16
SI-15 Information Output Filtering (P0)	0	0
SI-16 Memory Protection (P1)	1	1

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

Scan Summary - OWASP Mobile Top 10 2016

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

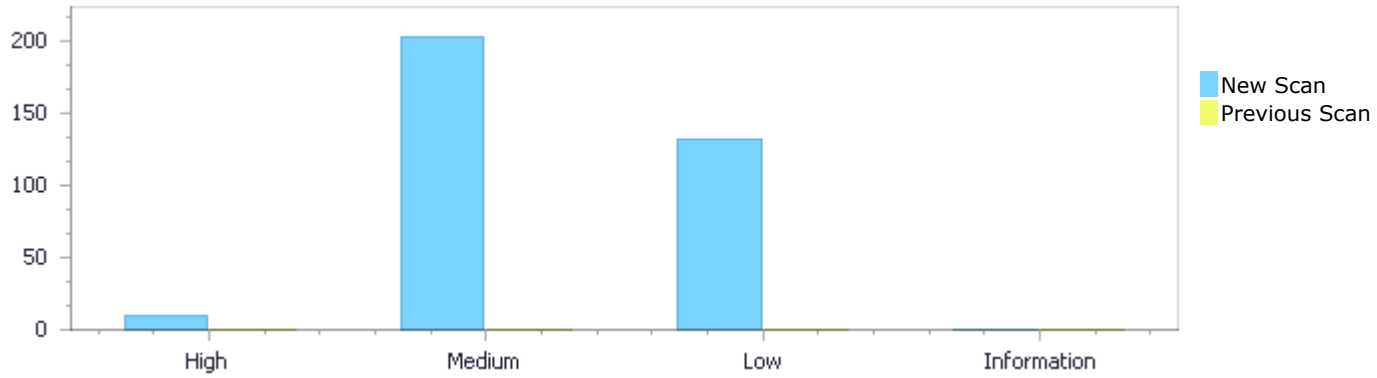
	modify the application's data and resources. This can provide the attacker a direct method of subverting the intended use of the software for personal or monetary gain.		
M9-Reverse Engineering	This category includes analysis of the final core binary to determine its source code, libraries, algorithms,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

Scan Summary - Custom

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

Results Distribution By Status First scan of the project

	High	Medium	Low	Information	Total
New Issues	10	203	132	0	345
Recurrent Issues	0	0	0	0	0
Total	10	203	132	0	345
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	10	203	132	0	345
Urgent	0	0	0	0	0
Proposed Not Exploitable	0	0	0	0	0
Total	10	203	132	0	345

Result Summary

Vulnerability Type	Occurrences	Severity
Buffer Overflow OutOfBound	6	High
Buffer Overflow IndexFromInput	3	High
Buffer Overflow boundedcpy	1	High
Dangerous Functions	77	Medium
Use of Zero Initialized Pointer	29	Medium

Memory Leak	28	Medium
MemoryFree on StackVariable	24	Medium
Buffer Overflow boundcpy WrongSizeParam	20	Medium
Wrong Size t Allocation	11	Medium
Integer Overflow	4	Medium
Heap Inspection	2	Medium
Inadequate Encryption Strength	2	Medium
Use of Uninitialized Pointer	2	Medium
Boolean Overflow	1	Medium
Divide By Zero	1	Medium
Double Free	1	Medium
Use of Uninitialized Variable	1	Medium
Improper Resource Access Authorization	30	Low
NULL Pointer Dereference	29	Low
Unchecked Array Index	25	Low
Unchecked Return Value	16	Low
Sizeof Pointer Argument	13	Low
Incorrect Permission Assignment For Critical Resources	6	Low
TOCTOU	5	Low
Use of Sizeof On a Pointer Type	4	Low
Exposure of System Data to Unauthorized Control Sphere	1	Low
Information Exposure Through Comments	1	Low
Insecure Temporary File	1	Low
Unreleased Resource Leak	1	Low

10 Most Vulnerable Files

High and Medium Vulnerabilities

File Name	Issues Found
AliOS-Things-2/url.c	49
AliOS-Things-2/quickjs-libc.c	38
AliOS-Things-2/cookie.c	32
AliOS-Things-2/nss.c	16
AliOS-Things-2/ch395_lwip.c	13
AliOS-Things-2/rtsp.c	11
AliOS-Things-2/smtp.c	10
AliOS-Things-2/l2cap.c	9
AliOS-Things-2/krb5.c	7
AliOS-Things-2/SDL_wave.c	7

Scan Results Details

Buffer Overflow OutOfBound

Query Path:
CPP\Cx\CPP Buffer Overflow\Buffer Overflow OutOfBound 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 OutOfBound\Path 1:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=1
Status	New

The size of the buffer used by js_printf_internal in q, at line 148 of AliOS-Things-2/quickjs-libc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that js_printf_internal passes to q, at line 148 of AliOS-Things-2/quickjs-libc.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	285	285
Object	q	q

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_printf_internal(JSContext *ctx,

```
....
285.                    q[2] = q[-1];
```

Buffer Overflow OutOfBound\Path 2:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=2
Status	New

The size of the buffer used by js_printf_internal in q, at line 148 of AliOS-Things-2/quickjs-libc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that js_printf_internal passes to q, at line 148 of AliOS-Things-2/quickjs-libc.c, to overwrite the target buffer.

	Source	Destination

File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	286	286
Object	q	q

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_printf_internal(JSContext *ctx,

```
....
286.                    q[-1] = 'I';
```

Buffer Overflow OutOfBound\Path 3:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=3
Status	New

The size of the buffer used by TIFFFetchByteArray in v, at line 496 of AliOS-Things-2/tif_pdsdirread.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that TIFFFetchShortPair passes to v, at line 549 of AliOS-Things-2/tif_pdsdirread.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/tif_pdsdirread.c	AliOS-Things-2/tif_pdsdirread.c
Line	551	505
Object	v	v

Code Snippet
File Name AliOS-Things-2/tif_pdsdirread.c
Method TIFFFetchShortPair(TIFF* tif, TIFFDirEntry* dir)

```
....
551.        uint16 v[2];
```

File Name AliOS-Things-2/tif_pdsdirread.c

Method TIFFFetchByteArray(TIFF* tif, TIFFDirEntry* dir, uint16* v)

```
....
505.                    case 3: v[2] = (dir->tdir_offset >> 8) & 0xff;
```

Buffer Overflow OutOfBound\Path 4:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=4

Status	New

The size of the buffer used by TIFFFetchByteArray in v, at line 496 of AliOS-Things-2/tif_pdsdirread.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that TIFFFetchShortPair passes to v, at line 549 of AliOS-Things-2/tif_pdsdirread.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/tif_pdsdirread.c	AliOS-Things-2/tif_pdsdirread.c
Line	551	504
Object	v	v

Code Snippet
File Name AliOS-Things-2/tif_pdsdirread.c
Method TIFFFetchShortPair(TIFF* tif, TIFFDirEntry* dir)

```
....
551.        uint16 v[2];
```

File Name AliOS-Things-2/tif_pdsdirread.c
Method TIFFFetchByteArray(TIFF* tif, TIFFDirEntry* dir, uint16* v)

```
....
504.                    case 4: v[3] = dir->tdir_offset & 0xff;
```

Buffer Overflow OutOfBound\Path 5:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=5
Status	New

The size of the buffer used by TIFFFetchByteArray in v, at line 496 of AliOS-Things-2/tif_pdsdirread.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that TIFFFetchShortPair passes to v, at line 549 of AliOS-Things-2/tif_pdsdirread.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/tif_pdsdirread.c	AliOS-Things-2/tif_pdsdirread.c
Line	551	511
Object	v	v

Code Snippet
File Name AliOS-Things-2/tif_pdsdirread.c
Method TIFFFetchShortPair(TIFF* tif, TIFFDirEntry* dir)

```
....
551.        uint16 v[2];
```

File Name AliOS-Things-2/tif_pdsdirread.c

Method TIFFFetchByteArray(TIFF* tif, TIFFDirEntry* dir, uint16* v)

```
....
511.                    case 4: v[3] = dir->tdir_offset >> 24;
```

Buffer Overflow OutOfBound\Path 6:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=6
Status	New

The size of the buffer used by TIFFFetchByteArray in v, at line 496 of AliOS-Things-2/tif_pdsdirread.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that TIFFFetchShortPair passes to v, at line 549 of AliOS-Things-2/tif_pdsdirread.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/tif_pdsdirread.c	AliOS-Things-2/tif_pdsdirread.c
Line	551	512
Object	v	v

Code Snippet

File Name AliOS-Things-2/tif_pdsdirread.c

Method TIFFFetchShortPair(TIFF* tif, TIFFDirEntry* dir)

```
....
551.        uint16 v[2];
```

File Name AliOS-Things-2/tif_pdsdirread.c

Method TIFFFetchByteArray(TIFF* tif, TIFFDirEntry* dir, uint16* v)

```
....
512.                    case 3: v[2] = (dir->tdir_offset >> 16) & 0xff;
```

Buffer Overflow IndexFromInput

Query Path:

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

Categories

OWASP Top 10 2017: A1-Injection

Description

Buffer Overflow IndexFromInput\Path 1:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=343
Status	New

The size of the buffer used by my_execvpe in BinaryExpr, at line 2661 of AliOS-Things-2/quickjs-libc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that my_execvpe passes to getenv, at line 2661 of AliOS-Things-2/quickjs-libc.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	2676	2696
Object	getenv	BinaryExpr

Code Snippet

File Name AliOS-Things-2/quickjs-libc.c

Method static int my_execvpe(const char *filename, char **argv, char **envp)

```
....
2676.       path = getenv("PATH");
....
2696.           buf[path_len + 1 + filename_len] = '\0';
```

Buffer Overflow IndexFromInput\Path 2:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=344
Status	New

The size of the buffer used by my_execvpe in path_len, at line 2661 of AliOS-Things-2/quickjs-libc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that my_execvpe passes to getenv, at line 2661 of AliOS-Things-2/quickjs-libc.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	2676	2694
Object	getenv	path_len

Code Snippet

File Name AliOS-Things-2/quickjs-libc.c

Method static int my_execvpe(const char *filename, char **argv, char **envp)

```
....
2676.       path = getenv("PATH");
....
2694.           buf[path_len] = '/';
```

Buffer Overflow IndexFromInput\Path 3:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=345
Status	New

The size of the buffer used by js_os_readlink in res, at line 2578 of AliOS-Things-2/quickjs-libc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that js_os_readlink passes to buf, at line 2578 of AliOS-Things-2/quickjs-libc.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	2589	2594
Object	buf	res

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_os_readlink(JSContext *ctx, JSValueConst this_val,

```
....
2589.       res = readlink(path, buf, sizeof(buf) - 1);
....
2594.           buf[res] = '\0';
```

Buffer Overflow boundedcpy

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

Categories

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

Description

Buffer Overflow boundedcpy\Path 1:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=73
Status	New

The size parameter path_len in line 2661 in file AliOS-Things-2/quickjs-libc.c is influenced by the user input getenv in line 2661 in file AliOS-Things-2/quickjs-libc.c. This may lead to a buffer overflow vulnerability, which may in turn result in malicious code execution.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	2676	2693
Object	getenv	path_len

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static int my_execvpe(const char *filename, char **argv, char **envp)

```
....
2676.       path = getenv("PATH");
....
2693.           memcpy(buf, p, path_len);
```

Dangerous Functions

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

Categories

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

Description
Dangerous Functions\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=158
Status	New

The dangerous function, memcpy, was found in use at line 67 in AliOS-Things-2/ch395_lwip.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/ch395_lwip.c	AliOS-Things-2/ch395_lwip.c
Line	93	93
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/ch395_lwip.c
Method static err_t low_level_output(struct netif *netif, struct pbuf *p)

```
....
93.           memcpy(&data[datalen], src_buf, copylen);
```

Dangerous Functions\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=159
Status	New

The dangerous function, memcpy, was found in use at line 122 in AliOS-Things-2/ch395_lwip.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/ch395_lwip.c	AliOS-Things-2/ch395_lwip.c
Line	155	155
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/ch395_lwip.c
Method static struct pbuf *low_level_input(struct netif *netif, uint8_t *data, uint32_t datalen)

```
....
155.          memcpy(dest_buf, &buffer[bufferoffset], copylen);
```

Dangerous Functions\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=160
Status	New

The dangerous function, memcpy, was found in use at line 396 in AliOS-Things-2/ch395_lwip.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/ch395_lwip.c	AliOS-Things-2/ch395_lwip.c
Line	415	415
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/ch395_lwip.c
Method static void tcpip_dhcpc_cb(struct netif *pstnetif)

```
....
415.              memcpy(ip_addr, &eth_ip_info.ip.addr,
sizeof(ip_addr));
```

Dangerous Functions\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=161
Status	New

The dangerous function, memcpy, was found in use at line 396 in AliOS-Things-2/ch395_lwip.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/ch395_lwip.c	AliOS-Things-2/ch395_lwip.c
Line	416	416
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/ch395_lwip.c
Method static void tcpip_dhcpc_cb(struct netif *pstnetif)

```
....
416.              memcpy(mask_addr, &eth_ip_info.netmask.addr,
sizeof(ip_addr));
```

Dangerous Functions\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=162
Status	New

The dangerous function, memcpy, was found in use at line 396 in AliOS-Things-2/ch395_lwip.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/ch395_lwip.c	AliOS-Things-2/ch395_lwip.c
Line	417	417
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/ch395_lwip.c
Method static void tcpip_dhcpc_cb(struct netif *pstnetif)

```
....
417.              memcpy(gw_addr, &eth_ip_info.gw.addr,
sizeof(ip_addr));
```

Dangerous Functions\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=163
Status	New

The dangerous function, memcpy, was found in use at line 541 in AliOS-Things-2/ch395_lwip.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/ch395_lwip.c	AliOS-Things-2/ch395_lwip.c
Line	638	638
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/ch395_lwip.c
Method static void ch395_lwip_inter_proc(void)

```
....
638.                            memcpy(&ipaddr.addr,
gst_lwipch395info.ip_info.ipaddr, 4);
```

Dangerous Functions\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=164
Status	New

The dangerous function, memcpy, was found in use at line 541 in AliOS-Things-2/ch395_lwip.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/ch395_lwip.c	AliOS-Things-2/ch395_lwip.c
Line	639	639
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/ch395_lwip.c
Method static void ch395_lwip_inter_proc(void)

```
....
639.                            memcpy(&netmask.addr,
gst_lwipch395info.ip_info.ip_mask, 4);
```

Dangerous Functions\Path 8:

Severity	Medium

Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=165
Status	New

The dangerous function, memcpy, was found in use at line 541 in AliOS-Things-2/ch395_lwip.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/ch395_lwip.c	AliOS-Things-2/ch395_lwip.c
Line	640	640
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/ch395_lwip.c
Method static void ch395_lwip_inter_proc(void)

```
....
640.                            memcpy(&gw.addr,
gst_lwipch395info.ip_info.gateway, 4);
```

Dangerous Functions\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=166
Status	New

The dangerous function, memcpy, was found in use at line 427 in AliOS-Things-2/cookie.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	755	755
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/cookie.c
Method Curl_cookie_add(struct Curl_easy *data,

```
....
755.            memcpy(co->path, path, pathlen);
```

Dangerous Functions\Path 10:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=167
Status	New

The dangerous function, memcpy, was found in use at line 113 in AliOS-Things-2/krb5.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/krb5.c	AliOS-Things-2/krb5.c
Line	139	139
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/krb5.c
Method krb5_encode(void *app_data, const void *from, int length, int level, void **to)

```
....
139.    memcpy(*to, enc.value, enc.length);
```

Dangerous Functions\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=168
Status	New

The dangerous function, memcpy, was found in use at line 2035 in AliOS-Things-2/l2cap.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	2055	2055
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/l2cap.c
Method int bt_l2cap_chan_recv_complete(struct bt_l2cap_chan *chan, struct net_buf *buf)

```
....
2055.       memcpy(&credits, net_buf_user_data(buf), sizeof(credits));
```

Dangerous Functions\Path 12:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=169

Status	New

The dangerous function, memcpy, was found in use at line 2101 in AliOS-Things-2/l2cap.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	2109	2109
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/l2cap.c
Method static void l2cap_chan_le_recv_seg(struct bt_l2cap_le_chan *chan,

```
....
2109.             memcpy(&seg, net_buf_user_data(chan->_sdu),
sizeof(seg));
```

Dangerous Functions\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=170
Status	New

The dangerous function, memcpy, was found in use at line 2101 in AliOS-Things-2/l2cap.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	2120	2120
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/l2cap.c
Method static void l2cap_chan_le_recv_seg(struct bt_l2cap_le_chan *chan,

```
....
2120.       memcpy(net_buf_user_data(chan->_sdu), &seg, sizeof(seg));
```

Dangerous Functions\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=171
Status	New

The dangerous function, memcpy, was found in use at line 1352 in AliOS-Things-2/nss.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	1388	1388
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_init(struct Curl_easy *data)

```
....
1388.       memcpy(&nspr_io_methods, PR_GetDefaultIOMethods(),
```

Dangerous Functions\Path 15:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=172
Status New

The dangerous function, memcpy, was found in use at line 1802 in AliOS-Things-2/nss.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	2067	2067
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_setup_connect(struct connectdata *conn, int sockindex)

```
....
2067.         memcpy(&protocols[cur], NGHTTP2_PROTO_VERSION_ID,
```

Dangerous Functions\Path 16:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=173
Status New

The dangerous function, memcpy, was found in use at line 1802 in AliOS-Things-2/nss.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	2073	2073
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_setup_connect(struct connectdata *conn, int sockindex)

```
....
2073.       memcpy(&protocols[cur], ALPN_HTTP_1_1, ALPN_HTTP_1_1_LENGTH);
```

Dangerous Functions\Path 17:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=174
Status New

The dangerous function, memcpy, was found in use at line 2601 in AliOS-Things-2/quickjs-libc.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	2637	2637
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static char **build_envp(JSContext *ctx, JSValueConst obj)

```
....
2637.           memcpy(pair, key, key_len);
```

Dangerous Functions\Path 18:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=175
Status New

The dangerous function, memcpy, was found in use at line 2601 in AliOS-Things-2/quickjs-libc.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c

Line	2639	2639
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static char **build_envp(JSContext *ctx, JSValueConst obj)

```
....
2639.           memcpy(pair + key_len + 1, str, str_len);
```

Dangerous Functions\Path 19:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=176
Status	New

The dangerous function, memcpy, was found in use at line 2661 in AliOS-Things-2/quickjs-libc.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	2693	2693
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static int my_execvpe(const char *filename, char **argv, char **envp)

```
....
2693.           memcpy(buf, p, path_len);
```

Dangerous Functions\Path 20:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=177
Status	New

The dangerous function, memcpy, was found in use at line 2661 in AliOS-Things-2/quickjs-libc.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	2695	2695
Object	memcpy	memcpy

Code Snippet	
File Name	AliOS-Things-2/quickjs-libc.c
Method	static int my_execvpe(const char *filename, char **argv, char **envp)

```
....
2695.          memcpy(buf + path_len + 1, filename, filename_len);
```

Dangerous Functions\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=178
Status	New

The dangerous function, memcpy, was found in use at line 3250 in AliOS-Things-2/quickjs-libc.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3284	3284
Object	memcpy	memcpy

Code Snippet	
File Name	AliOS-Things-2/quickjs-libc.c
Method	static JSValue js_worker_ctor(JSContext *ctx, JSValueConst new_target,

```
....
3284.      memcpy(args->eval_buf, str, str_len + 1);
```

Dangerous Functions\Path 22:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=179
Status	New

The dangerous function, memcpy, was found in use at line 3324 in AliOS-Things-2/quickjs-libc.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3353	3353
Object	memcpy	memcpy

Code Snippet	
File Name	AliOS-Things-2/quickjs-libc.c

Method	static JSValue js_worker_postMessage(JSContext *ctx, JSValueConst this_val,

```
....
3353.       memcpy(msg->data, data, data_len);
```

Dangerous Functions\Path 23:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=180
Status	New

The dangerous function, memcpy, was found in use at line 765 in AliOS-Things-2/rtsp.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/rtsp.c	AliOS-Things-2/rtsp.c
Line	820	820
Object	memcpy	memcpy

Code Snippet	
File Name	AliOS-Things-2/rtsp.c
Method	CURLcode Curl_rtsp_parseheader(struct connectdata *conn,

```
....
820.        memcpy(data->set.str[STRING_RTSP_SESSION_ID], start, end -
start);
```

Dangerous Functions\Path 24:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=181
Status	New

The dangerous function, memcpy, was found in use at line 598 in AliOS-Things-2/rtsp.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/rtsp.c	AliOS-Things-2/rtsp.c
Line	620	620
Object	memcpy	memcpy

Code Snippet	
File Name	AliOS-Things-2/rtsp.c
Method	static CURLcode rtsp_rtp_readwrite(struct Curl_easy *data,

```
....
620.      memcpy(rtspc->rtp_buf + rtspc->rtp_bufsize, k->str, *nread);
```

Dangerous Functions\Path 25:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=182
Status	New

The dangerous function, memcpy, was found in use at line 598 in AliOS-Things-2/rtsp.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/rtsp.c	AliOS-Things-2/rtsp.c
Line	692	692
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/rtsp.c
Method static CURLcode rtsp_rtp_readwrite(struct Curl_easy *data,

```
....
692.      memcpy(scratch, rtp, rtp_dataleft);
```

Dangerous Functions\Path 26:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=183
Status	New

The dangerous function, memcpy, was found in use at line 1543 in AliOS-Things-2/smtp.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/smtp.c	AliOS-Things-2/smtp.c
Line	1590	1590
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/smtp.c
Method CURLcode Curl_smtp_escape_eob(struct connectdata *conn, const ssize_t nread)

```
....
1590.         memcpy(&scratch[si], &SMTP_EOB[eob_sent], smtp->eob -
eob_sent);
```

Dangerous Functions\Path 27:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=184
Status	New

The dangerous function, memcpy, was found in use at line 1543 in AliOS-Things-2/smtp.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/smtp.c	AliOS-Things-2/smtp.c
Line	1608	1608
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/smtp.c
Method CURLcode Curl_smtp_escape_eob(struct connectdata *conn, const ssize_t nread)

```
....
1608.         memcpy(&scratch[si], &SMTP_EOB_REPL[eob_sent],
```

Dangerous Functions\Path 28:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=185
Status	New

The dangerous function, memcpy, was found in use at line 1543 in AliOS-Things-2/smtp.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/smtp.c	AliOS-Things-2/smtp.c
Line	1620	1620
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/smtp.c
Method CURLcode Curl_smtp_escape_eob(struct connectdata *conn, const ssize_t nread)

```
....
1620.      memcpy(&scratch[si], &SMTP_EOB[eob_sent], smtp->eob -
eob_sent);
```

Dangerous Functions\Path 29:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=186
Status	New

The dangerous function, memcpy, was found in use at line 196 in AliOS-Things-2/smtp.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/smtp.c	AliOS-Things-2/smtp.c
Line	215	215
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/smtp.c
Method static bool smtp_endofresp(struct connectdata *conn, char *line, size_t len,

```
....
215.      memcpy(tmpline, line, (len == 5 ? 5 : 3));
```

Dangerous Functions\Path 30:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=187
Status	New

The dangerous function, memcpy, was found in use at line 59 in AliOS-Things-2/tiff2dib.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/tiff2dib.c	AliOS-Things-2/tiff2dib.c
Line	191	191
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/tiff2dib.c
Method HDIB LoadTIFFinDIB(LPSTR lpFileName)

```
....
191.                                    memcpy(lpBits, &buf[(int)
(l*LineSize)], (int)
```

Dangerous Functions\Path 31:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=188
Status	New

The dangerous function, memcpy, was found in use at line 2667 in AliOS-Things-2/url.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2739	2739
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/url.c
Method CURLcode Curl_parse_login_details(const char *login, const size_t len,

```
....
2739.        memcpy(ubuf, login, ulen);
```

Dangerous Functions\Path 32:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=189
Status	New

The dangerous function, memcpy, was found in use at line 2667 in AliOS-Things-2/url.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2747	2747
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/url.c
Method CURLcode Curl_parse_login_details(const char *login, const size_t len,

```
....
2747.         memcpy(pbuf, psep + 1, plen);
```

Dangerous Functions\Path 33:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=190
Status	New

The dangerous function, memcpy, was found in use at line 2667 in AliOS-Things-2/url.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2755	2755
Object	memcpy	memcpy

Code Snippet
File Name AliOS-Things-2/url.c
Method CURLcode Curl_parse_login_details(const char *login, const size_t len,

```
....
2755.         memcpy(obuf, osep + 1, olen);
```

Dangerous Functions\Path 34:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=191
Status	New

The dangerous function, sscanf, was found in use at line 765 in AliOS-Things-2/rtsp.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/rtsp.c	AliOS-Things-2/rtsp.c
Line	773	773
Object	sscanf	sscanf

Code Snippet
File Name AliOS-Things-2/rtsp.c
Method CURLcode Curl_rtsp_parseheader(struct connectdata *conn,

```
....
773.      int nc = sscanf(&header[4], ": %ld", &CSeq);
```

Dangerous Functions\Path 35:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=192
Status	New

The dangerous function, strcpy, was found in use at line 525 in AliOS-Things-2/quickjs-libc.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	543	543
Object	strcpy	strcpy

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method int js_module_set_import_meta(JSContext *ctx, JSValueConst func_val,

```
....
543.          strcpy(buf, "file://");
```

Dangerous Functions\Path 36:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=193
Status	New

The dangerous function, strcpy, was found in use at line 2223 in AliOS-Things-2/url.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2254	2254
Object	strcpy	strcpy

Code Snippet
File Name AliOS-Things-2/url.c
Method static char *detect_proxy(struct connectdata *conn)

```
....
2254.    strcpy(envp, "_proxy");
```

Dangerous Functions\Path 37:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=194
Status	New

The dangerous function, strlen, was found in use at line 118 in AliOS-Things-2/cookie.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	120	120
Object	strlen	strlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static bool tailmatch(const char *cooke_domain, const char *hostname)

```
....
120.    size_t cookie_domain_len = strlen(cooke_domain);
```

Dangerous Functions\Path 38:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=195
Status	New

The dangerous function, strlen, was found in use at line 118 in AliOS-Things-2/cookie.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	121	121
Object	strlen	strlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static bool tailmatch(const char *cooke_domain, const char *hostname)

```
....
121.    size_t hostname_len = strlen(hostname);
```

Dangerous Functions\Path 39:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=196
Status	New

The dangerous function, strlen, was found in use at line 169 in AliOS-Things-2/cookie.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	178	178
Object	strlen	strlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static bool pathmatch(const char *cookie_path, const char *request_uri)

```
....
178.    cookie_path_len = strlen(cookie_path);
```

Dangerous Functions\Path 40:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=197
Status	New

The dangerous function, strlen, was found in use at line 169 in AliOS-Things-2/cookie.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	192	192
Object	strlen	strlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static bool pathmatch(const char *cookie_path, const char *request_uri)

```
....
192.    if(0 == strlen(uri_path) || uri_path[0] != '/') {
```

Dangerous Functions\Path 41:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=198
Status	New

The dangerous function, strlen, was found in use at line 169 in AliOS-Things-2/cookie.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	208	208
Object	strlen	strlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static bool pathmatch(const char *cookie_path, const char *request_uri)

```
....
208.    uri_path_len = strlen(uri_path);
```

Dangerous Functions\Path 42:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=199
Status	New

The dangerous function, strlen, was found in use at line 243 in AliOS-Things-2/cookie.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	251	251
Object	strlen	strlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static const char *get_top_domain(const char * const domain, size_t *outlen)

```
....
251.    len = strlen(domain);
```

Dangerous Functions\Path 43:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=200
Status	New

The dangerous function, strlen, was found in use at line 299 in AliOS-Things-2/cookie.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	307	307
Object	strlen	strlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static char *sanitize_cookie_path(const char *cookie_path)

```
....
307.    len = strlen(new_path);
```

Dangerous Functions\Path 44:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=201
Status	New

The dangerous function, strlen, was found in use at line 427 in AliOS-Things-2/cookie.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	466	466
Object	strlen	strlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method Curl_cookie_add(struct Curl_easy *data,

```
....
466.      size_t linelength = strlen(lineptr);
```

Dangerous Functions\Path 45:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=202
Status	New

The dangerous function, strlen, was found in use at line 427 in AliOS-Things-2/cookie.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	491	491
Object	strlen	strlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method Curl_cookie_add(struct Curl_easy *data,

```
....
491.          size_t len = strlen(what);
```

Dangerous Functions\Path 46:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=203
Status	New

The dangerous function, strlen, was found in use at line 427 in AliOS-Things-2/cookie.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	492	492
Object	strlen	strlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method Curl_cookie_add(struct Curl_easy *data,

```
....
492.          size_t nlen = strlen(name);
```

Dangerous Functions\Path 47:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=204
Status	New

The dangerous function, strlen, was found in use at line 427 in AliOS-Things-2/cookie.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1008	1008
Object	strlen	strlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method Curl_cookie_add(struct Curl_easy *data,

```
....
1008.                cllen = strlen(clist->spath);
```

Dangerous Functions\Path 48:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=205
Status	New

The dangerous function, strlen, was found in use at line 1186 in AliOS-Things-2/cookie.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1193	1193
Object	strlen	strlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static int cookie_sort(const void *p1, const void *p2)

```
....
1193.    l1 = c1->path ? strlen(c1->path) : 0;
```

Dangerous Functions\Path 49:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=206
Status	New

The dangerous function, strlen, was found in use at line 1186 in AliOS-Things-2/cookie.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1194	1194
Object	strlen	strlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static int cookie_sort(const void *p1, const void *p2)

```
....
1194.    l2 = c2->path ? strlen(c2->path) : 0;
```

Dangerous Functions\Path 50:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=207
Status	New

The dangerous function, strlen, was found in use at line 1186 in AliOS-Things-2/cookie.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1200	1200
Object	strlen	strlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static int cookie_sort(const void *p1, const void *p2)

```
....
1200.    l1 = c1->domain ? strlen(c1->domain) : 0;
```

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-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=269
Status	New

The variable declared in tok_buf at AliOS-Things-2/cookie.c in line 427 is not initialized when it is used by lastc at AliOS-Things-2/cookie.c in line 427.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	779	1068
Object	tok_buf	lastc

Code Snippet
File Name AliOS-Things-2/cookie.c
Method Curl_cookie_add(struct Curl_easy *data,

```
....
779.      char *tok_buf = NULL;
....
1068.      lastc = clist;
```

Use of Zero Initialized Pointer\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=270
Status	New

The variable declared in tok_buf at AliOS-Things-2/cookie.c in line 427 is not initialized when it is used by lastc at AliOS-Things-2/cookie.c in line 427.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c

Line	779	1062
Object	tok_buf	lastc

Code Snippet
File Name AliOS-Things-2/cookie.c
Method Curl_cookie_add(struct Curl_easy *data,

```
....
779.      char *tok_buf = NULL;
....
1062.            lastc = clist;
```

Use of Zero Initialized Pointer\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=271
Status	New

The variable declared in tok_buf at AliOS-Things-2/cookie.c in line 427 is not initialized when it is used by cookies at AliOS-Things-2/cookie.c in line 378.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	779	386
Object	tok_buf	cookies

Code Snippet
File Name AliOS-Things-2/cookie.c
Method Curl_cookie_add(struct Curl_easy *data,

```
....
779.      char *tok_buf = NULL;
```

File Name AliOS-Things-2/cookie.c
Method static void remove_expired(struct CookieInfo *cookies)

```
....
386.      co = cookies->cookies[i];
```

Use of Zero Initialized Pointer\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=272
Status	New

The variable declared in tok_buf at AliOS-Things-2/cookie.c in line 427 is not initialized when it is used by cookies at AliOS-Things-2/cookie.c in line 427.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	779	973
Object	tok_buf	cookies

Code Snippet
File Name AliOS-Things-2/cookie.c
Method Curl_cookie_add(struct Curl_easy *data,

```
....
779.      char *tok_buf = NULL;
....
973.    clist = c->cookies[myhash];
```

Use of Zero Initialized Pointer\Path 5:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=273
Status New

The variable declared in tok_buf at AliOS-Things-2/cookie.c in line 427 is not initialized when it is used by cookies at AliOS-Things-2/cookie.c in line 427.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	779	1084
Object	tok_buf	cookies

Code Snippet
File Name AliOS-Things-2/cookie.c
Method Curl_cookie_add(struct Curl_easy *data,

```
....
779.      char *tok_buf = NULL;
....
1084.        c->cookies[myhash] = co;
```

Use of Zero Initialized Pointer\Path 6:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=274
Status New

The variable declared in tok_buf at AliOS-Things-2/cookie.c in line 427 is not initialized when it is used by first at AliOS-Things-2/cookie.c in line 243.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	779	254
Object	tok_buf	first

Code Snippet
File Name AliOS-Things-2/cookie.c
Method Curl_cookie_add(struct Curl_easy *data,

```
....
779.      char *tok_buf = NULL;
```

File Name AliOS-Things-2/cookie.c
Method static const char *get_top_domain(const char * const domain, size_t *outlen)

```
....
254.      first = memrchr(domain, '.', (last - domain));
```

Use of Zero Initialized Pointer\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=275
Status	New

The variable declared in mainco at AliOS-Things-2/cookie.c in line 1273 is not initialized when it is used by mainco at AliOS-Things-2/cookie.c in line 1273.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1279	1319
Object	mainco	mainco

Code Snippet
File Name AliOS-Things-2/cookie.c
Method struct Cookie *Curl_cookie_getlist(struct CookieInfo *c,

```
....
1279.    struct Cookie *mainco = NULL;
....
1319.              mainco = newco;
```

Use of Zero Initialized Pointer\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=276
Status	New

The variable declared in mainco at AliOS-Things-2/cookie.c in line 1273 is not initialized when it is used by mainco at AliOS-Things-2/cookie.c in line 1273.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1279	1353
Object	mainco	mainco

Code Snippet
File Name AliOS-Things-2/cookie.c
Method struct Cookie *Curl_cookie_getlist(struct CookieInfo *c,

```
....
1279.    struct Cookie *mainco = NULL;
....
1353.      mainco = array[0]; /* start here */
```

Use of Zero Initialized Pointer\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=277
Status	New

The variable declared in list at AliOS-Things-2/cookie.c in line 1581 is not initialized when it is used by list at AliOS-Things-2/cookie.c in line 1581.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1583	1608
Object	list	list

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static struct curl_slist *cookie_list(struct Curl_easy *data)

```
....
1583.    struct curl_slist *list = NULL;
....
1608.        list = beg;
```

Use of Zero Initialized Pointer\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=278
Status	New

The variable declared in ace_hostname at AliOS-Things-2/url.c in line 1598 is not initialized when it is used by ace_hostname at AliOS-Things-2/url.c in line 1598.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	1617	1628
Object	ace_hostname	ace_hostname

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode idnconvert_hostname(struct connectdata *conn,

```
....
1617.         char *ace_hostname = NULL;
....
1628.           host->encalloc = (char *)ace_hostname;
```

Use of Zero Initialized Pointer\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=279
Status	New

The variable declared in psep at AliOS-Things-2/url.c in line 2667 is not initialized when it is used by psep at AliOS-Things-2/url.c in line 2667.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2687	2747
Object	psep	psep

Code Snippet
File Name AliOS-Things-2/url.c
Method CURLcode Curl_parse_login_details(const char *login, const size_t len,

```
....
2687.         psep = NULL;
....
2747.         memcpy(pbuf, psep + 1, plen);
```

Use of Zero Initialized Pointer\Path 12:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=280
Status	New

The variable declared in psep at AliOS-Things-2/url.c in line 2667 is not initialized when it is used by psep at AliOS-Things-2/url.c in line 2667.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2675	2747
Object	psep	psep

Code Snippet
File Name AliOS-Things-2/url.c
Method CURLcode Curl_parse_login_details(const char *login, const size_t len,

```
....
2675.    const char *psep = NULL;
....
2747.        memcpy(pbuf, psep + 1, plen);
```

Use of Zero Initialized Pointer\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=281
Status	New

The variable declared in osep at AliOS-Things-2/url.c in line 2667 is not initialized when it is used by osep at AliOS-Things-2/url.c in line 2667.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2696	2755
Object	osep	osep

Code Snippet
File Name AliOS-Things-2/url.c
Method CURLcode Curl_parse_login_details(const char *login, const size_t len,

```
....
2696.        osep = NULL;
....
2755.        memcpy(obuf, osep + 1, olen);
```

Use of Zero Initialized Pointer\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=282
Status	New

The variable declared in osep at AliOS-Things-2/url.c in line 2667 is not initialized when it is used by osep at AliOS-Things-2/url.c in line 2667.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2676	2755
Object	osep	osep

Code Snippet
File Name AliOS-Things-2/url.c
Method CURLcode Curl_parse_login_details(const char *login, const size_t len,

```
....
2676.    const char *osep = NULL;
....
2755.        memcpy(obuf, osep + 1, olen);
```

Use of Zero Initialized Pointer\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=283
Status	New

The variable declared in passwd at AliOS-Things-2/url.c in line 2469 is not initialized when it is used by hostname_resolve at AliOS-Things-2/url.c in line 3181.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2607	3250
Object	passwd	hostname_resolve

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode create_conn_helper_init_proxy(struct connectdata *conn)

```
....
2607.            conn->http_proxy.passwd = NULL;
```

File Name	AliOS-Things-2/url.c
Method	static CURLcode resolve_server(struct Curl_easy *data,

```
....
3250.        conn->hostname_resolve = strdup(connhost->name);
```

Use of Zero Initialized Pointer\Path 16:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=284
Status	New

The variable declared in user at AliOS-Things-2/url.c in line 2469 is not initialized when it is used by hostname_resolve at AliOS-Things-2/url.c in line 3181.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2604	3250
Object	user	hostname_resolve

Code Snippet

File Name	AliOS-Things-2/url.c
Method	static CURLcode create_conn_helper_init_proxy(struct connectdata *conn)

```
....
2604.            conn->http_proxy.user = NULL;
```

File Name	AliOS-Things-2/url.c
Method	static CURLcode resolve_server(struct Curl_easy *data,

```
....
3250.        conn->hostname_resolve = strdup(connhost->name);
```

Use of Zero Initialized Pointer\Path 17:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=285
Status	New

The variable declared in endp at AliOS-Things-2/url.c in line 2935 is not initialized when it is used by hostname_resolve at AliOS-Things-2/url.c in line 3181.

	Source	Destination

File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3001	3250
Object	endp	hostname_resolve

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode parse_connect_to_host_port(struct Curl_easy *data,

```
....
3001.       char *endp = NULL;
```

File Name AliOS-Things-2/url.c
Method static CURLcode resolve_server(struct Curl_easy *data,

```
....
3250.         conn->hostname_resolve = strdup(connhost->name);
```

Use of Zero Initialized Pointer\Path 18:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=286
Status	New

The variable declared in cert_dir at AliOS-Things-2/nss.c in line 1352 is not initialized when it is used by nss_context at AliOS-Things-2/nss.c in line 1308.

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	1369	1326
Object	cert_dir	nss_context

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_init(struct Curl_easy *data)

```
....
1369.         cert_dir = NULL;
```

File Name AliOS-Things-2/nss.c
Method static CURLcode nss_init_core(struct Curl_easy *data, const char *cert_dir)

```
....
1326.       nss_context = NSS_InitContext(certpath, "", "", "",
&initparams,
```

Use of Zero Initialized Pointer\Path 19:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=287
Status	New

The variable declared in cafile at AliOS-Things-2/nss.c in line 1592 is not initialized when it is used by cafile at AliOS-Things-2/nss.c in line 1592.

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	1603	1612
Object	cafile	cafile

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_load_ca_certificates(struct connectdata *conn,

```
....
1603.       cafile = NULL;
....
1612.    use_trust_module = !cafile && !capath;
```

Use of Zero Initialized Pointer\Path 20:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=288
Status	New

The variable declared in capath at AliOS-Things-2/nss.c in line 1592 is not initialized when it is used by capath at AliOS-Things-2/nss.c in line 1592.

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	1605	1612
Object	capath	capath

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_load_ca_certificates(struct connectdata *conn,

```
....
1605.       capath = NULL;
....
1612.    use_trust_module = !cafile && !capath;
```

Use of Zero Initialized Pointer\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=289
Status	New

The variable declared in sab_tab at AliOS-Things-2/quickjs-libc.c in line 3324 is not initialized when it is used by sab_tab at AliOS-Things-2/quickjs-libc.c in line 3324.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3347	3367
Object	sab_tab	sab_tab

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_worker_postMessage(JSContext *ctx, JSValueConst this_val,

```
....
3347.       msg->sab_tab = NULL;
....
3367.           js_sab_dup(NULL, msg->sab_tab[i]);
```

Use of Zero Initialized Pointer\Path 22:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=290
Status	New

The variable declared in ref at AliOS-Things-2/rtsp.c in line 235 is not initialized when it is used by p_referrer at AliOS-Things-2/rtsp.c in line 235.

	Source	Destination
File	AliOS-Things-2/rtsp.c	AliOS-Things-2/rtsp.c
Line	404	406
Object	ref	p_referrer

Code Snippet
File Name AliOS-Things-2/rtsp.c
Method static CURLcode rtsp_do(struct connectdata *conn, bool *done)

```
....
404.      conn->allocptr.ref = NULL;
....
406.    p_referrer = conn->allocptr.ref;
```

Use of Zero Initialized Pointer\Path 23:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=291
Status	New

The variable declared in Pointer at AliOS-Things-2/SDL_wave.c in line 1408 is not initialized when it is used by audio_buf at AliOS-Things-2/SDL_wave.c in line 2097.

	Source	Destination
File	AliOS-Things-2/SDL_wave.c	AliOS-Things-2/SDL_wave.c
Line	1423	2128
Object	Pointer	audio_buf

Code Snippet

File Name AliOS-Things-2/SDL_wave.c

Method PCM_Decode(WaveFile *file, Uint8 **audio_buf, Uint32 *audio_len)

```
....
1423.          *audio_buf = NULL;
```

File Name AliOS-Things-2/SDL_wave.c

Method SDL_LoadWAV_RW(SDL_RWops *src, int freesrc, SDL_AudioSpec *spec, Uint8 **audio_buf, Uint32 *audio_len)

```
....
2128.          SDL_free(*audio_buf);
```

Use of Zero Initialized Pointer\Path 24:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=292
Status	New

The variable declared in Pointer at AliOS-Things-2/SDL_wave.c in line 1170 is not initialized when it is used by audio_buf at AliOS-Things-2/SDL_wave.c in line 2097.

	Source	Destination

File	AliOS-Things-2/SDL_wave.c	AliOS-Things-2/SDL_wave.c
Line	1226	2128
Object	Pointer	audio_buf

Code Snippet
File Name AliOS-Things-2/SDL_wave.c
Method LAW_Decode(WaveFile *file, Uint8 **audio_buf, Uint32 *audio_len)

```
....
1226.           *audio_buf = NULL;
```

File Name AliOS-Things-2/SDL_wave.c

Method SDL_LoadWAV_RW(SDL_RWops *src, int freesrc, SDL_AudioSpec *spec, Uint8 **audio_buf, Uint32 *audio_len)

```
....
2128.           SDL_free(*audio_buf);
```

Use of Zero Initialized Pointer\Path 25:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=293
Status	New

The variable declared in Pointer at AliOS-Things-2/SDL_wave.c in line 1037 is not initialized when it is used by audio_buf at AliOS-Things-2/SDL_wave.c in line 2097.

	Source	Destination
File	AliOS-Things-2/SDL_wave.c	AliOS-Things-2/SDL_wave.c
Line	1054	2128
Object	Pointer	audio_buf

Code Snippet
File Name AliOS-Things-2/SDL_wave.c
Method IMA_ADPCM_Decode(WaveFile *file, Uint8 **audio_buf, Uint32 *audio_len)

```
....
1054.           *audio_buf = NULL;
```

File Name AliOS-Things-2/SDL_wave.c

Method SDL_LoadWAV_RW(SDL_RWops *src, int freesrc, SDL_AudioSpec *spec, Uint8 **audio_buf, Uint32 *audio_len)

```
....
2128.           SDL_free(*audio_buf);
```

Use of Zero Initialized Pointer\Path 26:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=294
Status	New

The variable declared in Pointer at AliOS-Things-2/SDL_wave.c in line 641 is not initialized when it is used by audio_buf at AliOS-Things-2/SDL_wave.c in line 2097.

	Source	Destination
File	AliOS-Things-2/SDL_wave.c	AliOS-Things-2/SDL_wave.c
Line	661	2128
Object	Pointer	audio_buf

Code Snippet

File Name AliOS-Things-2/SDL_wave.c

Method MS_ADPCM_Decode(WaveFile *file, Uint8 **audio_buf, Uint32 *audio_len)

```
....
661.           *audio_buf = NULL;
```

File Name AliOS-Things-2/SDL_wave.c

Method SDL_LoadWAV_RW(SDL_RWops *src, int freesrc, SDL_AudioSpec *spec, Uint8 **audio_buf, Uint32 *audio_len)

```
....
2128.           SDL_free(*audio_buf);
```

Use of Zero Initialized Pointer\Path 27:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=295
Status	New

The variable declared in Pointer at AliOS-Things-2/SDL_wave.c in line 2097 is not initialized when it is used by audio_buf at AliOS-Things-2/SDL_wave.c in line 2097.

	Source	Destination
File	AliOS-Things-2/SDL_wave.c	AliOS-Things-2/SDL_wave.c

Line	2119	2128
Object	Pointer	audio_buf

Code Snippet
File Name AliOS-Things-2/SDL_wave.c
Method SDL_LoadWAV_RW(SDL_RWops *src, int freesrc, SDL_AudioSpec *spec, Uint8 **audio_buf, Uint32 *audio_len)

```
....
2119.       *audio_buf = NULL;
....
2128.           SDL_free(*audio_buf);
```

Use of Zero Initialized Pointer\Path 28:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=296
Status New

The variable declared in data at AliOS-Things-2/SDL_wave.c in line 1511 is not initialized when it is used by data at AliOS-Things-2/SDL_wave.c in line 1554.

	Source	Destination
File	AliOS-Things-2/SDL_wave.c	AliOS-Things-2/SDL_wave.c
Line	1515	1563
Object	data	data

Code Snippet
File Name AliOS-Things-2/SDL_wave.c
Method WaveFreeChunkData(WaveChunk *chunk)

```
....
1515.           chunk->data = NULL;
```

File Name AliOS-Things-2/SDL_wave.c
Method WaveReadPartialChunkData(SDL_RWops *src, WaveChunk *chunk, size_t length)

```
....
1563.           chunk->data = SDL_malloc(length);
```

Use of Zero Initialized Pointer\Path 29:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=297

Status	New

The variable declared in hostaddr at AliOS-Things-2/url.c in line 3181 is not initialized when it is used by dns_entry at AliOS-Things-2/url.c in line 3181.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3229	3293
Object	hostaddr	dns_entry

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode resolve_server(struct Curl_easy *data,

```
....
3229.             hostaddr = NULL;
....
3293.      conn->dns_entry = hostaddr;
```

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-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=238
Status	New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	1951	1951
Object	nickname	nickname

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_setup_connect(struct connectdata *conn, int sockindex)

```
....
1951.      char *nickname = dup_nickname(data, SSL_SET_OPTION(cert));
```

Memory Leak\Path 2:

Severity	Medium

Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=239
Status	New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	429	429
Object	wrap	wrap

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode insert_wrapped_ptr(struct curl_llist *list, void *ptr)

```
....
429.    struct ptr_list_wrap *wrap = malloc(sizeof(*wrap));
```

Memory Leak\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=240
Status	New

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1118	1118
Object	filename	filename

Code Snippet
File Name AliOS-Things-2/cookie.c
Method struct CookieInfo *Curl_cookie_init(struct Curl_easy *data,

```
....
1118.      c->filename = strdup(file?file:"none"); /* copy the name just
in case */
```

Memory Leak\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=241
Status	New

	Source	Destination

File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	2367	2367
Object	f	f

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_os_readdir(JSContext *ctx, JSValueConst this_val,

```
....
2367.       f = opendir(path);
```

Memory Leak\Path 5:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=242
Status New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3053	3053
Object	sab	sab

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static void *js_sab_alloc(void *opaque, size_t size)

```
....
3053.       sab = malloc(sizeof(JSSABHeader) + size);
```

Memory Leak\Path 6:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=243
Status New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3087	3087
Object	ps	ps

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSWorkerMessagePipe *js_new_message_pipe(void)

```
....
3087.       ps = malloc(sizeof(*ps));
```

Memory Leak\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=244
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3279	3279
Object	eval_buf	eval_buf

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_worker_ctor(JSContext *ctx, JSValueConst new_target,

```
....
3279.       args->eval_buf = malloc(str_len + 1);
```

Memory Leak\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=245
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3350	3350
Object	data	data

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_worker_postMessage(JSContext *ctx, JSValueConst this_val,

```
....
3350.       msg->data = malloc(data_len);
```

Memory Leak\Path 9:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=246
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3662	3662
Object	ts	ts

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method void js_std_init_handlers(JSRuntime *rt)

```
....
3662.       ts = malloc(sizeof(*ts));
```

Memory Leak\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=247
Status	New

	Source	Destination
File	AliOS-Things-2/rtsp.c	AliOS-Things-2/rtsp.c
Line	122	122
Object	rtsp	rtsp

Code Snippet
File Name AliOS-Things-2/rtsp.c
Method static CURLcode rtsp_setup_connection(struct connectdata *conn)

```
....
122.    conn->data->req.protop = rtsp = calloc(1, sizeof(struct RTSP));
```

Memory Leak\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=248
Status	New

	Source	Destination
File	AliOS-Things-2/rtsp.c	AliOS-Things-2/rtsp.c
Line	685	685

Object	scratch	scratch

Code Snippet
File Name AliOS-Things-2/rtsp.c
Method static CURLcode rtsp_rtp_readwrite(struct Curl_easy *data,

```
....
685.      scratch = malloc(rtp_dataleft);
```

Memory Leak\Path 12:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=249
Status New

	Source	Destination
File	AliOS-Things-2/smtp.c	AliOS-Things-2/smtp.c
Line	1114	1114
Object	protop	protop

Code Snippet
File Name AliOS-Things-2/smtp.c
Method static CURLcode smtp_init(struct connectdata *conn)

```
....
1114.    smtp = data->req.protop = calloc(sizeof(struct SMTP), 1);
```

Memory Leak\Path 13:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=250
Status New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	1951	1951
Object	user	user

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode parseurlandfillconn(struct Curl_easy *data,

```
....
1951.       conn->user = strdup(data->state.up.user);
```

Memory Leak\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=251
Status	New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	1962	1962
Object	passwd	passwd

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode parseurlandfillconn(struct Curl_easy *data,

```
....
1962.       conn->passwd = strdup(data->state.up.password);
```

Memory Leak\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=252
Status	New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	1973	1973
Object	options	options

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode parseurlandfillconn(struct Curl_easy *data,

```
....
1973.       conn->options = strdup(data->state.up.options);
```

Memory Leak\Path 16:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=253
Status	New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2046	2046
Object	rawalloc	rawalloc

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode parseurlandfillconn(struct Curl_easy *data,

```
....
2046.    conn->host.rawalloc = strdup(hostname);
```

Memory Leak\Path 17:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=254
Status	New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2074	2074
Object	range	range

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode setup_range(struct Curl_easy *data)

```
....
2074.        s->range = strdup(data->set.str[STRING_SET_RANGE]);
```

Memory Leak\Path 18:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=255
Status	New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2382	2382

Object	proxypasswd	proxypasswd

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode parse_proxy(struct Curl_easy *data,

```
....
2382.        proxypasswd = strdup("");
```

Memory Leak\Path 19:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=256
Status New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2728	2728
Object	obuf	obuf

Code Snippet
File Name AliOS-Things-2/url.c
Method CURLcode Curl_parse_login_details(const char *login, const size_t len,

```
....
2728.      obuf = malloc(olen + 1);
```

Memory Leak\Path 20:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=257
Status New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3250	3250
Object	hostname_resolve	hostname_resolve

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode resolve_server(struct Curl_easy *data,

```
....
3250.         conn->hostname_resolve = strdup(connhost->name);
```

Memory Leak\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=258
Status	New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3274	3274
Object	hostname_resolve	hostname_resolve

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode resolve_server(struct Curl_easy *data,

```
....
3274.         conn->hostname_resolve = strdup(host->name);
```

Memory Leak\Path 22:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=259
Status	New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3450	3450
Object	oauth_bearer	oauth_bearer

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3450.       conn->oauth_bearer = strdup(data->set.str[STRING_BEARER]);
```

Memory Leak\Path 23:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=260
Status	New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3459	3459
Object	unix_domain_socket	unix_domain_socket

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3459.      conn->unix_domain_socket = strdup(data-
>set.str[STRING_UNIX_SOCKET_PATH]);
```

Memory Leak\Path 24:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=261
Status	New

	Source	Destination
File	AliOS-Things-2/krb5.c	AliOS-Things-2/krb5.c
Line	136	136
Object	to	to

Code Snippet
File Name AliOS-Things-2/krb5.c
Method krb5_encode(void *app_data, const void *from, int length, int level, void **to)

```
....
136.    *to = malloc(enc.length);
```

Memory Leak\Path 25:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=262
Status	New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c

Line	2818	2818
Object	userp	userp

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode override_login(struct Curl_easy *data,

```
....
2818.       *userp = strdup(data->set.str[STRING_USERNAME]);
```

Memory Leak\Path 26:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=263
Status New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2827	2827
Object	passwdp	passwdp

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode override_login(struct Curl_easy *data,

```
....
2827.       *passwdp = strdup(data->set.str[STRING_PASSWORD]);
```

Memory Leak\Path 27:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=264
Status New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2836	2836
Object	optionsp	optionsp

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode override_login(struct Curl_easy *data,

```
....
2836.       *optionsp = strdup(data->set.str[STRING_OPTIONS]);
```

Memory Leak\Path 28:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=265
Status	New

	Source	Destination
File	AliOS-Things-2/rtsp.c	AliOS-Things-2/rtsp.c
Line	817	817
Object	str	str

Code Snippet
File Name AliOS-Things-2/rtsp.c
Method CURLcode Curl_rtsp_parseheader(struct connectdata *conn,

```
....
817.        data->set.str[STRING_RTSP_SESSION_ID] = malloc(end - start +
1);
```

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-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=49
Status	New

Calling free() (line 1581) on a variable that was not dynamically allocated (line 1581) in file AliOS-Things-2/cookie.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1604	1604
Object	line	line

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static struct curl_slist *cookie_list(struct Curl_easy *data)

```
....
1604.          free(line);
```

MemoryFree on StackVariable\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=50
Status	New

Calling free() (line 411) on a variable that was not dynamically allocated (line 411) in file AliOS-Things-2/curl_sasl.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/curl_sasl.c	AliOS-Things-2/curl_sasl.c
Line	477	477
Object	chlg	chlg

Code Snippet
File Name AliOS-Things-2/curl_sasl.c
Method CURLcode Curl_sasl_continue(struct SASL *sasl, struct connectdata *conn,

```
....
477.      free(chlg);
```

MemoryFree on StackVariable\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=51
Status	New

Calling free() (line 256) on a variable that was not dynamically allocated (line 256) in file AliOS-Things-2/curl_sasl.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/curl_sasl.c	AliOS-Things-2/curl_sasl.c
Line	390	390
Object	resp	resp

Code Snippet
File Name AliOS-Things-2/curl_sasl.c
Method CURLcode Curl_sasl_start(struct SASL *sasl, struct connectdata *conn,

```
....
390.        free(resp);
```

MemoryFree on StackVariable\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=52
Status	New

Calling free() (line 256) on a variable that was not dynamically allocated (line 256) in file AliOS-Things-2/curl_sasl.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/curl_sasl.c	AliOS-Things-2/curl_sasl.c
Line	401	401
Object	resp	resp

Code Snippet
File Name AliOS-Things-2/curl_sasl.c
Method CURLcode Curl_sasl_start(struct SASL *sasl, struct connectdata *conn,

```
....
401.    free(resp);
```

MemoryFree on StackVariable\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=53
Status	New

Calling free() (line 146) on a variable that was not dynamically allocated (line 146) in file AliOS-Things-2/krb5.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/krb5.c	AliOS-Things-2/krb5.c
Line	205	205
Object	stringp	stringp

Code Snippet
File Name AliOS-Things-2/krb5.c
Method krb5_auth(void *app_data, struct connectdata *conn)

```
....
205.        free(stringp);
```

MemoryFree on StackVariable\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=54
Status	New

Calling free() (line 146) on a variable that was not dynamically allocated (line 146) in file AliOS-Things-2/krb5.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/krb5.c	AliOS-Things-2/krb5.c
Line	267	267
Object	p	p

Code Snippet
File Name AliOS-Things-2/krb5.c
Method krb5_auth(void *app_data, struct connectdata *conn)

```
....
267.            free(p);
```

MemoryFree on StackVariable\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=55
Status	New

Calling free() (line 146) on a variable that was not dynamically allocated (line 146) in file AliOS-Things-2/krb5.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/krb5.c	AliOS-Things-2/krb5.c
Line	268	268
Object	cmd	cmd

Code Snippet
File Name AliOS-Things-2/krb5.c
Method krb5_auth(void *app_data, struct connectdata *conn)

```
....
268.          free(cmd);
```

MemoryFree on StackVariable\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=56
Status	New

Calling free() (line 441) on a variable that was not dynamically allocated (line 441) in file AliOS-Things-2/nss.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	461	461
Object	slot_name	slot_name

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_create_object(struct ssl_connect_data *connssl,

```
....
461.    free(slot_name);
```

MemoryFree on StackVariable\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=57
Status	New

Calling free() (line 505) on a variable that was not dynamically allocated (line 505) in file AliOS-Things-2/nss.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	511	511
Object	wrap	wrap

Code Snippet
File Name AliOS-Things-2/nss.c
Method static void nss_destroy_object(void *user, void *ptr)

```
....
511.    free(wrap);
```

MemoryFree on StackVariable\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=58
Status	New

Calling free() (line 515) on a variable that was not dynamically allocated (line 515) in file AliOS-Things-2/nss.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	521	521
Object	wrap	wrap

Code Snippet
File Name AliOS-Things-2/nss.c
Method static void nss_destroy_crl_item(void *user, void *ptr)

```
....
521.    free(wrap);
```

MemoryFree on StackVariable\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=59
Status	New

Calling free() (line 524) on a variable that was not dynamically allocated (line 524) in file AliOS-Things-2/nss.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	554	554
Object	nickname	nickname

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_load_cert(struct ssl_connect_data *ssl,

```
....
554.        free(nickname);
```

MemoryFree on StackVariable\Path 12:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=60
Status	New

Calling free() (line 1264) on a variable that was not dynamically allocated (line 1264) in file AliOS-Things-2/nss.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	1278	1278
Object	config_string	config_string

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_load_module(SECMODModule **pmod, const char *library,

```
....
1278.    free(config_string);
```

MemoryFree on StackVariable\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=61
Status	New

Calling free() (line 1308) on a variable that was not dynamically allocated (line 1308) in file AliOS-Things-2/nss.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	1328	1328
Object	certpath	certpath

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_init_core(struct Curl_easy *data, const char *cert_dir)

```
....
1328.       free(certpath);
```

MemoryFree on StackVariable\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=62
Status	New

Calling free() (line 1592) on a variable that was not dynamically allocated (line 1592) in file AliOS-Things-2/nss.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	1661	1661
Object	fullpath	fullpath

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_load_ca_certificates(struct connectdata *conn,

```
....
1661.           free(fullpath);
```

MemoryFree on StackVariable\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=63
Status	New

Calling free() (line 3060) on a variable that was not dynamically allocated (line 3060) in file AliOS-Things-2/quickjs-libc.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3068	3068
Object	sab	sab

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static void js_sab_free(void *opaque, void *ptr)

```
....
3068.           free(sab);
```

MemoryFree on StackVariable\Path 16:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=64
Status	New

Calling free() (line 3168) on a variable that was not dynamically allocated (line 3168) in file AliOS-Things-2/quickjs-libc.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3204	3204
Object	args	args

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static void *worker_func(void *opaque)

```
....
3204.       free(args);
```

MemoryFree on StackVariable\Path 17:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=65
Status	New

Calling free() (line 3688) on a variable that was not dynamically allocated (line 3688) in file AliOS-Things-2/quickjs-libc.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3710	3710
Object	ts	ts

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method void js_std_free_handlers(JSRuntime *rt)

```
....
3710.       free(ts);
```

MemoryFree on StackVariable\Path 18:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=66
Status	New

Calling free() (line 231) on a variable that was not dynamically allocated (line 231) in file AliOS-Things-2/SDL_wave.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/SDL_wave.c	AliOS-Things-2/SDL_wave.c
Line	317	317
Object	dumpstr	dumpstr

Code Snippet

File Name AliOS-Things-2/SDL_wave.c

Method WaveDebugDumpFormat(WaveFile *file, Uint32 rifflen, Uint32 fmtlen, Uint32 datalen)

```
....
317.       free(dumpstr);
```

MemoryFree on StackVariable\Path 19:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=67
Status	New

Calling free() (line 1543) on a variable that was not dynamically allocated (line 1543) in file AliOS-Things-2/smtp.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/smtp.c	AliOS-Things-2/smtp.c
Line	1633	1633
Object	oldscratch	oldscratch

Code Snippet

File Name AliOS-Things-2/smtp.c

Method CURLcode Curl_smtp_escape_eob(struct connectdata *conn, const ssize_t nread)

```
....
1633.       free(oldscratch);
```

MemoryFree on StackVariable\Path 20:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=68
Status	New

Calling free() (line 511) on a variable that was not dynamically allocated (line 511) in file AliOS-Things-2/smtp.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/smtp.c	AliOS-Things-2/smtp.c
Line	606	606
Object	size	size

Code Snippet
File Name AliOS-Things-2/smtp.c
Method static CURLcode smtp_perform_mail(struct connectdata *conn)

```
....
606.     free(size);
```

MemoryFree on StackVariable\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=69
Status	New

Calling free() (line 1893) on a variable that was not dynamically allocated (line 1893) in file AliOS-Things-2/url.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2041	2041
Object	zoneid	zoneid

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode parseurlandfillconn(struct Curl_easy *data,

```
....
2041.        free(zoneid);
```

MemoryFree on StackVariable\Path 22:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=70
Status	New

Calling free() (line 2301) on a variable that was not dynamically allocated (line 2301) in file AliOS-Things-2/url.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2396	2396
Object	portptr	portptr

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode parse_proxy(struct Curl_easy *data,

```
....
2396.       free(portptr);
```

MemoryFree on StackVariable\Path 23:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=71
Status	New

Calling free() (line 2301) on a variable that was not dynamically allocated (line 2301) in file AliOS-Things-2/url.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2433	2433
Object	scheme	scheme

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode parse_proxy(struct Curl_easy *data,

```
....
2433.     free(scheme);
```

MemoryFree on StackVariable\Path 24:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=72
Status	New

Calling free() (line 3036) on a variable that was not dynamically allocated (line 3036) in file AliOS-Things-2/url.c may result with a crash.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3067	3067
Object	hostname_to_match	hostname_to_match

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode parse_connect_to_string(struct Curl_easy *data,

```
....
3067.       free(hostname_to_match);
```

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-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=28
Status	New

The size of the buffer used by tcpip_dhcpc_cb in ip_addr, at line 396 of AliOS-Things-2/ch395_lwip.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that tcpip_dhcpc_cb passes to ip_addr, at line 396 of AliOS-Things-2/ch395_lwip.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/ch395_lwip.c	AliOS-Things-2/ch395_lwip.c

Line	416	416
Object	ip_addr	ip_addr

Code Snippet
File Name AliOS-Things-2/ch395_lwip.c
Method static void tcpip_dhcpc_cb(struct netif *pstnetif)

```
....
416.                memcpy(mask_addr, &eth_ip_info.netmask.addr,
sizeof(ip_addr));
```

Buffer Overflow boundcpy WrongSizeParam\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=29
Status	New

The size of the buffer used by tcpip_dhcpc_cb in ip_addr, at line 396 of AliOS-Things-2/ch395_lwip.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that tcpip_dhcpc_cb passes to ip_addr, at line 396 of AliOS-Things-2/ch395_lwip.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/ch395_lwip.c	AliOS-Things-2/ch395_lwip.c
Line	417	417
Object	ip_addr	ip_addr

Code Snippet
File Name AliOS-Things-2/ch395_lwip.c
Method static void tcpip_dhcpc_cb(struct netif *pstnetif)

```
....
417.                memcpy(gw_addr, &eth_ip_info.gw.addr,
sizeof(ip_addr));
```

Buffer Overflow boundcpy WrongSizeParam\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=30
Status	New

The size of the buffer used by l2cap_chan_le_recv_seg in seg, at line 2101 of AliOS-Things-2/l2cap.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that l2cap_chan_le_recv_seg passes to seg, at line 2101 of AliOS-Things-2/l2cap.c, to overwrite the target buffer.

	Source	Destination

File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	2120	2120
Object	seg	seg

Code Snippet
File Name AliOS-Things-2/l2cap.c
Method static void l2cap_chan_le_recv_seg(struct bt_l2cap_le_chan *chan,

```
....
2120.       memcpy(net_buf_user_data(chan->_sdu), &seg, sizeof(seg));
```

Buffer Overflow boundcpy WrongSizeParam\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=31
Status	New

The size of the buffer used by ch395_lwip_inter_proc in Namespace974666809, at line 541 of AliOS-Things-2/ch395_lwip.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ch395_lwip_inter_proc passes to Namespace974666809, at line 541 of AliOS-Things-2/ch395_lwip.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/ch395_lwip.c	AliOS-Things-2/ch395_lwip.c
Line	621	621
Object	Namespace974666809	Namespace974666809

Code Snippet
File Name AliOS-Things-2/ch395_lwip.c
Method static void ch395_lwip_inter_proc(void)

```
....
621.                    memset(&gst_lwipch395info.ip_info, 0,
sizeof(gst_lwipch395info.ip_info));
```

Buffer Overflow boundcpy WrongSizeParam\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=32
Status	New

The size of the buffer used by l2cap_chan_tx_init in ->, at line 851 of AliOS-Things-2/l2cap.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that l2cap_chan_tx_init passes to ->, at line 851 of AliOS-Things-2/l2cap.c, to overwrite the target buffer.

	Source	Destination

File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	855	855
Object	->	->

Code Snippet
File Name AliOS-Things-2/l2cap.c
Method static void l2cap_chan_tx_init(struct bt_l2cap_le_chan *chan)

```
....
855.        (void)memset(&chan->tx, 0, sizeof(chan->tx));
```

Buffer Overflow boundcpy WrongSizeParam\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=33
Status	New

The size of the buffer used by nss_init_core in initparams, at line 1308 of AliOS-Things-2/nss.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that nss_init_core passes to initparams, at line 1308 of AliOS-Things-2/nss.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	1317	1317
Object	initparams	initparams

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_init_core(struct Curl_easy *data, const char *cert_dir)

```
....
1317.    memset((void *) &initparams, '\0', sizeof(initparams));
```

Buffer Overflow boundcpy WrongSizeParam\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=34
Status	New

The size of the buffer used by Curl_connect in SingleRequest, at line 3945 of AliOS-Things-2/url.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that Curl_connect passes to SingleRequest, at line 3945 of AliOS-Things-2/url.c, to overwrite the target buffer.

	Source	Destination

File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3956	3956
Object	SingleRequest	SingleRequest

Code Snippet
File Name AliOS-Things-2/url.c
Method CURLcode Curl_connect(struct Curl_easy *data,

```
....
3956.    memset(&data->req, 0, sizeof(struct SingleRequest));
```

Buffer Overflow boundcpy WrongSizeParam\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=35
Status	New

The size of the buffer used by le_ecred_conn_req in i, at line 1068 of AliOS-Things-2/l2cap.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that le_ecred_conn_req passes to i, at line 1068 of AliOS-Things-2/l2cap.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	1156	1156
Object	i	i

Code Snippet
File Name AliOS-Things-2/l2cap.c
Method static void le_ecred_conn_req(struct bt_l2cap *l2cap, u8_t ident,

```
....
1156.            memset(dcid, 0, sizeof(scid) * i);
```

Buffer Overflow boundcpy WrongSizeParam\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=36
Status	New

The size of the buffer used by le_ecred_conn_req in scid, at line 1068 of AliOS-Things-2/l2cap.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that le_ecred_conn_req passes to scid, at line 1068 of AliOS-Things-2/l2cap.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c

Line	1156	1156
Object	scid	scid

Code Snippet
File Name AliOS-Things-2/l2cap.c
Method static void le_ecred_conn_req(struct bt_l2cap *l2cap, u8_t ident,

```
....
1156.             memset(dcid, 0, sizeof(scid) * i);
```

Buffer Overflow boundcpy WrongSizeParam\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=37
Status	New

The size of the buffer used by low_level_output in copylen, at line 67 of AliOS-Things-2/ch395_lwip.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that low_level_output passes to copylen, at line 67 of AliOS-Things-2/ch395_lwip.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/ch395_lwip.c	AliOS-Things-2/ch395_lwip.c
Line	93	93
Object	copylen	copylen

Code Snippet
File Name AliOS-Things-2/ch395_lwip.c
Method static err_t low_level_output(struct netif *netif, struct pbuf *p)

```
....
93.           memcpy(&data[datalen], src_buf, copylen);
```

Buffer Overflow boundcpy WrongSizeParam\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=38
Status	New

The size of the buffer used by *low_level_input in copylen, at line 122 of AliOS-Things-2/ch395_lwip.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *low_level_input passes to copylen, at line 122 of AliOS-Things-2/ch395_lwip.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/ch395_lwip.c	AliOS-Things-2/ch395_lwip.c

Line	155	155
Object	copylen	copylen

Code Snippet
File Name AliOS-Things-2/ch395_lwip.c
Method static struct pbuf *low_level_input(struct netif *netif, uint8_t *data, uint32_t datalen)

```
....
155.          memcpy(dest_buf, &buffer[bufferoffset], copylen);
```

Buffer Overflow boundcpy WrongSizeParam\Path 12:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=39
Status	New

The size of the buffer used by krb5_encode in enc, at line 113 of AliOS-Things-2/krb5.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that krb5_encode passes to enc, at line 113 of AliOS-Things-2/krb5.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/krb5.c	AliOS-Things-2/krb5.c
Line	139	139
Object	enc	enc

Code Snippet
File Name AliOS-Things-2/krb5.c
Method krb5_encode(void *app_data, const void *from, int length, int level, void **to)

```
....
139.    memcpy(*to, enc.value, enc.length);
```

Buffer Overflow boundcpy WrongSizeParam\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=40
Status	New

The size of the buffer used by **build_envp in key_len, at line 2601 of AliOS-Things-2/quickjs-libc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that **build_envp passes to key_len, at line 2601 of AliOS-Things-2/quickjs-libc.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c

Line	2637	2637
Object	key_len	key_len

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static char **build_envp(JSContext *ctx, JSValueConst obj)

```
....
2637.           memcpy(pair, key, key_len);
```

Buffer Overflow boundcpy WrongSizeParam\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=41
Status	New

The size of the buffer used by **build_envp in str_len, at line 2601 of AliOS-Things-2/quickjs-libc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that **build_envp passes to str_len, at line 2601 of AliOS-Things-2/quickjs-libc.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	2639	2639
Object	str_len	str_len

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static char **build_envp(JSContext *ctx, JSValueConst obj)

```
....
2639.           memcpy(pair + key_len + 1, str, str_len);
```

Buffer Overflow boundcpy WrongSizeParam\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=42
Status	New

The size of the buffer used by my_execvpe in path_len, at line 2661 of AliOS-Things-2/quickjs-libc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that my_execvpe passes to path_len, at line 2661 of AliOS-Things-2/quickjs-libc.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c

Line	2693	2693
Object	path_len	path_len

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static int my_execvpe(const char *filename, char **argv, char **envp)

```
....
2693.           memcpy(buf, p, path_len);
```

Buffer Overflow boundcpy WrongSizeParam\Path 16:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=43
Status	New

The size of the buffer used by my_execvpe in filename_len, at line 2661 of AliOS-Things-2/quickjs-libc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that my_execvpe passes to filename_len, at line 2661 of AliOS-Things-2/quickjs-libc.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	2695	2695
Object	filename_len	filename_len

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static int my_execvpe(const char *filename, char **argv, char **envp)

```
....
2695.           memcpy(buf + path_len + 1, filename, filename_len);
```

Buffer Overflow boundcpy WrongSizeParam\Path 17:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=44
Status	New

The size of the buffer used by js_worker_postMessage in data_len, at line 3324 of AliOS-Things-2/quickjs-libc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that js_worker_postMessage passes to data_len, at line 3324 of AliOS-Things-2/quickjs-libc.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c

Line	3353	3353
Object	data_len	data_len

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_worker_postMessage(JSContext *ctx, JSValueConst this_val,

```
....
3353.       memcpy(msg->data, data, data_len);
```

Buffer Overflow boundcpy WrongSizeParam\Path 18:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=45
Status	New

The size of the buffer used by *sanitize_cookie_path in len, at line 299 of AliOS-Things-2/cookie.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that *sanitize_cookie_path passes to len, at line 299 of AliOS-Things-2/cookie.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	309	309
Object	len	len

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static char *sanitize_cookie_path(const char *cookie_path)

```
....
309.       memmove((void *)new_path, (const void *)(new_path + 1), len);
```

Buffer Overflow boundcpy WrongSizeParam\Path 19:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=46
Status	New

The size of the buffer used by parse_proxy_auth in MAX_CURL_USER_LENGTH, at line 2441 of AliOS-Things-2/url.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_proxy_auth passes to MAX_CURL_USER_LENGTH, at line 2441 of AliOS-Things-2/url.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c

Line	2450	2450
Object	MAX_CURL_USER_LENGTH	MAX_CURL_USER_LENGTH

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode parse_proxy_auth(struct Curl_easy *data,

```
....
2450.               MAX_CURL_USER_LENGTH);
```

Buffer Overflow boundcpy WrongSizeParam\Path 20:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=47
Status	New

The size of the buffer used by parse_proxy_auth in MAX_CURL_PASSWORD_LENGTH, at line 2441 of AliOS-Things-2/url.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that parse_proxy_auth passes to MAX_CURL_PASSWORD_LENGTH, at line 2441 of AliOS-Things-2/url.c, to overwrite the target buffer.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2455	2455
Object	MAX_CURL_PASSWORD_LENGTH	MAX_CURL_PASSWORD_LENGTH

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode parse_proxy_auth(struct Curl_easy *data,

```
....
2455.               MAX_CURL_PASSWORD_LENGTH);
```

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-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=74
Status	New

The function alloc in AliOS-Things-2/escape.c at line 79 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	AliOS-Things-2/escape.c	AliOS-Things-2/escape.c
Line	96	96
Object	alloc	alloc

Code Snippet
File Name AliOS-Things-2/escape.c
Method char *curl_easy_escape(struct Curl_easy *data, const char *string,

```
....
96.    ns = malloc(alloc);
```

Wrong Size t Allocation\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=75
Status	New

The function alloc in AliOS-Things-2/escape.c at line 147 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	AliOS-Things-2/escape.c	AliOS-Things-2/escape.c
Line	153	153
Object	alloc	alloc

Code Snippet
File Name AliOS-Things-2/escape.c
Method CURLcode Curl_urldecode(struct Curl_easy *data,

```
....
153.    char *ns = malloc(alloc);
```

Wrong Size t Allocation\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=76
Status	New

The function data_len in AliOS-Things-2/quickjs-libc.c at line 3324 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	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3350	3350
Object	data_len	data_len

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_worker_postMessage(JSContext *ctx, JSValueConst this_val,

```
....
3350.       msg->data = malloc(data_len);
```

Wrong Size t Allocation\Path 4:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=77
Status New

The function sslsize in AliOS-Things-2/url.c at line 1687 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	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	1700	1700
Object	sslsize	sslsize

Code Snippet
File Name AliOS-Things-2/url.c
Method static struct connectdata *allocate_conn(struct Curl_easy *data)

```
....
1700.       char *ssl = calloc(4, sslsize);
```

Wrong Size t Allocation\Path 5:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=78
Status New

The function pathlen in AliOS-Things-2/cookie.c at line 427 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	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	753	753

Object	pathlen	pathlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method Curl_cookie_add(struct Curl_easy *data,

```
....
753.          co->path = malloc(pathlen + 1); /* one extra for the zero
byte */
```

Wrong Size t Allocation\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=79
Status	New

The function matches in AliOS-Things-2/cookie.c at line 1273 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	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1339	1339
Object	matches	matches

Code Snippet
File Name AliOS-Things-2/cookie.c
Method struct Cookie *Curl_cookie_getlist(struct CookieInfo *c,

```
....
1339.      array = malloc(sizeof(struct Cookie *) * matches);
```

Wrong Size t Allocation\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=80
Status	New

The function buf_len in AliOS-Things-2/quickjs-libc.c at line 364 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	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	390	390

Object	buf_len	buf_len

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method uint8_t *js_load_file(JSContext *ctx, size_t *pbuf_len, const char *filename)

```
....
390.          buf = malloc(buf_len + 1);
```

Wrong Size t Allocation\Path 8:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=81
Status New

The function str_len in AliOS-Things-2/quickjs-libc.c at line 3250 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	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3279	3279
Object	str_len	str_len

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_worker_ctor(JSContext *ctx, JSValueConst new_target,

```
....
3279.      args->eval_buf = malloc(str_len + 1);
```

Wrong Size t Allocation\Path 9:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=82
Status New

The function ulen in AliOS-Things-2/url.c at line 2667 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	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2712	2712
Object	ulen	ulen

Code Snippet	
File Name	AliOS-Things-2/url.c
Method	CURLcode Curl_parse_login_details(const char *login, const size_t len,

```
....
2712.       ubuf = malloc(ulen + 1);
```

Wrong Size t Allocation\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=83
Status	New

The function plen in AliOS-Things-2/url.c at line 2667 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	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2719	2719
Object	plen	plen

Code Snippet	
File Name	AliOS-Things-2/url.c
Method	CURLcode Curl_parse_login_details(const char *login, const size_t len,

```
....
2719.       pbuf = malloc(plen + 1);
```

Wrong Size t Allocation\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=84
Status	New

The function olen in AliOS-Things-2/url.c at line 2667 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	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2728	2728
Object	olen	olen

Code Snippet	
File Name	AliOS-Things-2/url.c

Method	CURLcode Curl_parse_login_details(const char *login, const size_t len,

```
....
2728.      obuf = malloc(olen + 1);
```

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-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=115
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 59 of AliOS-Things-2/tiff2dib.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	AliOS-Things-2/tiff2dib.c	AliOS-Things-2/tiff2dib.c
Line	93	93
Object	AssignExpr	AssignExpr

Code Snippet	
File Name	AliOS-Things-2/tiff2dib.c
Method	HDIB LoadTIFFinDIB(LPSTR lpFileName)

```
....
93.      SamplePerPixel = (int) (LineSize/imageWidth);
```

Integer Overflow\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=116
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 59 of AliOS-Things-2/tiff2dib.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	AliOS-Things-2/tiff2dib.c	AliOS-Things-2/tiff2dib.c

Line	96	96
Object	AssignExpr	AssignExpr

Code Snippet
File Name AliOS-Things-2/tiff2dib.c
Method HDIB LoadTIFFinDIB(LPSTR lpFileName)

```
....
96.      Align = 4 - (LineSize % 4);
```

Integer Overflow\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=117
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 2935 of AliOS-Things-2/url.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3013	3013
Object	AssignExpr	AssignExpr

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode parse_connect_to_host_port(struct Curl_easy *data,

```
....
3013.          port = (int)portparse; /* we know it will fit */
```

Integer Overflow\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=118
Status	New

A variable of a larger data type, buffsize, is being assigned to a smaller data type, in 392 of AliOS-Things-2/mesalink.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	AliOS-Things-2/mesalink.c	AliOS-Things-2/mesalink.c
Line	397	397
Object	buffsize	buffsize

Code Snippet	
File Name	AliOS-Things-2/mesalink.c
Method	mesalink_recv(struct connectdata *conn, int num, char *buf, size_t buffersize,

```
....
397.    int buffsize = (buffersize > (size_t)INT_MAX) ? INT_MAX :
(int)buffersize;
```

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-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=236
Status	New

Method rtsp_do at line 235 of AliOS-Things-2/rtsp.c defines p_proxyuserpwd, which is designated to contain user passwords. However, while plaintext passwords are later assigned to p_proxyuserpwd, this variable is never cleared from memory.

	Source	Destination
File	AliOS-Things-2/rtsp.c	AliOS-Things-2/rtsp.c
Line	254	254
Object	p_proxyuserpwd	p_proxyuserpwd

Code Snippet	
File Name	AliOS-Things-2/rtsp.c
Method	static CURLcode rtsp_do(struct connectdata *conn, bool *done)

```
....
254.    const char *p_proxyuserpwd = NULL;
```

Heap Inspection\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=237
Status	New

Method rtsp_do at line 235 of AliOS-Things-2/rtsp.c defines p_userpwd, which is designated to contain user passwords. However, while plaintext passwords are later assigned to p_userpwd, this variable is never cleared from memory.

	Source	Destination
File	AliOS-Things-2/rtsp.c	AliOS-Things-2/rtsp.c
Line	255	255
Object	p_userpwd	p_userpwd

Code Snippet
File Name AliOS-Things-2/rtsp.c
Method static CURLcode rtsp_do(struct connectdata *conn, bool *done)

```
....
255.    const char *p_userpwd = NULL;
```

Use of Uninitialized Pointer

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

Categories

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

Description

Use of Uninitialized Pointer\Path 1:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=266
Status New

The variable declared in server at AliOS-Things-2/l2cap.c in line 715 is not initialized when it is used by server at AliOS-Things-2/l2cap.c in line 715.

	Source	Destination
File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	717	721
Object	server	server

Code Snippet
File Name AliOS-Things-2/l2cap.c
Method static struct bt_l2cap_server *l2cap_server_lookup_psm(u16_t psm)

```
....
717.        struct bt_l2cap_server *server;
....
721.                    return server;
```

Use of Uninitialized Pointer\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=267
Status	New

The variable declared in server at AliOS-Things-2/l2cap.c in line 715 is not initialized when it is used by psm at AliOS-Things-2/l2cap.c in line 715.

	Source	Destination
File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	717	720
Object	server	psm

Code Snippet
File Name AliOS-Things-2/l2cap.c
Method static struct bt_l2cap_server *l2cap_server_lookup_psm(u16_t psm)

```
....
717.        struct bt_l2cap_server *server;
....
720.                if (server->psm == psm) {
```

Inadequate Encryption Strength

Query Path:
CPP\Cx\CPP Medium Threat\Inadequate Encryption Strength Version:1

Categories

FISMA 2014: Configuration Management
NIST SP 800-53: SC-13 Cryptographic Protection (P1)
OWASP Top 10 2017: A3-Sensitive Data Exposure

Description
Inadequate Encryption Strength\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=298
Status	New

The application uses a weak cryptographic algorithm, Curl_auth_create_cram_md5_message at line 411 of AliOS-Things-2/curl_sasl.c, to protect sensitive personal information passwd, from AliOS-Things-2/curl_sasl.c at line 411.

	Source	Destination
File	AliOS-Things-2/curl_sasl.c	AliOS-Things-2/curl_sasl.c
Line	476	475
Object	passwd	Curl_auth_create_cram_md5_message

Code Snippet
File Name AliOS-Things-2/curl_sasl.c
Method CURLcode Curl_sasl_continue(struct SASL *sasl, struct connectdata *conn,

```
....
476.                                                    conn->passwd,
&resp, &len);
....
475.         result = Curl_auth_create_cram_md5_message(data, chlg, conn-
>user,
```

Inadequate Encryption Strength\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=299
Status	New

The application uses a weak cryptographic algorithm, Curl_auth_create_digest_md5_message at line 411 of AliOS-Things-2/curl_sasl.c, to protect sensitive personal information passwd, from AliOS-Things-2/curl_sasl.c at line 411.

	Source	Destination
File	AliOS-Things-2/curl_sasl.c	AliOS-Things-2/curl_sasl.c
Line	482	481
Object	passwd	Curl_auth_create_digest_md5_message

Code Snippet
File Name AliOS-Things-2/curl_sasl.c
Method CURLcode Curl_sasl_continue(struct SASL *sasl, struct connectdata *conn,

```
....
482.                                                    conn->user, conn-
>passwd,
....
481.       result = Curl_auth_create_digest_md5_message(data, serverdata,
```

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-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=27
Status	New

The application performs an illegal operation in ChopUpSingleUncompressedStrip, in AliOS-Things-2/tif_pdsdirread.c. In line 1060, the program attempts to divide by rowbytes, 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 rowbytes in ChopUpSingleUncompressedStrip of AliOS-Things-2/tif_pdsdirread.c, at line 1060.

	Source	Destination
File	AliOS-Things-2/tif_pdsdirread.c	AliOS-Things-2/tif_pdsdirread.c
Line	1078	1078
Object	rowbytes	rowbytes

Code Snippet
File Name AliOS-Things-2/tif_pdsdirread.c
Method ChopUpSingleUncompressedStrip(TIFF* tif)

```
....
1078.             rowsperstrip = 8192 / rowbytes;
```

Boolean Overflow

Query Path:
CPP\Cx\CPP Integer Overflow\Boolean 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
Boolean Overflow\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=114
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 894 of AliOS-Things-2/url.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	909	909
Object	AssignExpr	AssignExpr

Code Snippet
File Name AliOS-Things-2/url.c
Method static bool extract_if_dead(struct connectdata *conn,

```
....
909.        dead = (state & CONNRESULT_DEAD);
```

Double Free

Query Path:

CPP\Cx\CPP Medium Threat\Double Free Version:1

Categories

NIST SP 800-53: SI-16 Memory Protection (P1)

Description

Double Free\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=235
Status	New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2721	2731
Object	ubuf	ubuf

Code Snippet

File Name AliOS-Things-2/url.c

Method CURLcode Curl_parse_login_details(const char *login, const size_t len,

```
....
2721.         free(ubuf);
....
2731.         free(ubuf);
```

Use of Uninitialized Variable

Query Path:

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

Categories

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

Description

Use of Uninitialized Variable\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=268
Status	New

	Source	Destination
File	AliOS-Things-2/tiff2dib.c	AliOS-Things-2/tiff2dib.c
Line	74	112
Object	lpBits	lpBits

Code Snippet

File Name	AliOS-Things-2/tiff2dib.c
Method	HDIB LoadTIFFinDIB(LPSTR lpFileName)

```
....
74.      char          *lpBits;
....
112.      if (lpBits)
```

Improper Resource Access Authorization

Query Path:
CPP\Cx\CPP Low Visibility\Improper Resource Access Authorization Version:1

Categories

FISMA 2014: Identification And Authentication
NIST SP 800-53: AC-3 Access Enforcement (P1)
OWASP Top 10 2017: A2-Broken Authentication

Description

Improper Resource Access Authorization\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=300
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	1108	1108
Object	fgetc	fgetc

Code Snippet

File Name	AliOS-Things-2/quickjs-libc.c
Method	static JSValue js_std_file_getline(JSContext *ctx, JSValueConst this_val,

```
....
1108.          c = fgetc(f);
```

Improper Resource Access Authorization\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=301
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	1159	1159

Object	fgetc	fgetc

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_std_file_readAsString(JSContext *ctx, JSValueConst this_val,

```
....
1159.           c = fgetc(f);
```

Improper Resource Access Authorization\Path 3:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=302
Status New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	1179	1179
Object	fgetc	fgetc

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_std_file_getByte(JSContext *ctx, JSValueConst this_val,

```
....
1179.       return JS_NewInt32(ctx, fgetc(f));
```

Improper Resource Access Authorization\Path 4:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=303
Status New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	1208	1208
Object	fgetc	fgetc

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static int http_get_header_line(FILE *f, char *buf, size_t buf_size,

```
....
1208.           c = fgetc(f);
```

Improper Resource Access Authorization\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=304
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	393	393
Object	buf	buf

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method uint8_t *js_load_file(JSContext *ctx, size_t *pbuf_len, const char *filename)

```
....
393.      if (fread(buf, 1, buf_len, f) != buf_len) {
```

Improper Resource Access Authorization\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=305
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	1090	1090
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_std_file_read_write(JSContext *ctx, JSValueConst this_val,

```
....
1090.           ret = fread(buf + pos, 1, len, f);
```

Improper Resource Access Authorization\Path 7:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=306
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	1325	1325
Object	buf	buf

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_std_urlGet(JSContext *ctx, JSValueConst this_val,

```
....
1325.           len = fread(buf, 1, URL_GET_BUF_SIZE, f);
```

Improper Resource Access Authorization\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=307
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	1574	1574
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_os_read_write(JSContext *ctx, JSValueConst this_val,

```
....
1574.           ret = js_get_errno(read(fd, buf + pos, len));
```

Improper Resource Access Authorization\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=308
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	2589	2589

Object	buf	buf

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_os_readlink(JSContext *ctx, JSValueConst this_val,

```
....
2589.      res = readlink(path, buf, sizeof(buf) - 1);
```

Improper Resource Access Authorization\Path 10:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=309
Status New

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1563	1563
Object	fprintf	fprintf

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static int cookie_output(struct CookieInfo *c, const char *dumphere)

```
....
1563.          fprintf(out, "#\n# Fatal libcurl error\n");
```

Improper Resource Access Authorization\Path 11:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=310
Status New

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1569	1569
Object	fprintf	fprintf

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static int cookie_output(struct CookieInfo *c, const char *dumphere)

```
....
1569.        fprintf(out, "%s\n", format_ptr);
```

Improper Resource Access Authorization\Path 12:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=311
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3178	3178
Object	fprintf	fprintf

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static void *worker_func(void *opaque)

```
....
3178.          fprintf(stderr, "JS_NewRuntime failure");
```

Improper Resource Access Authorization\Path 13:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=312
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3190	3190
Object	fprintf	fprintf

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static void *worker_func(void *opaque)

```
....
3190.          fprintf(stderr, "JS_NewContext failure");
```

Improper Resource Access Authorization\Path 14:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=313
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3664	3664
Object	fprintf	fprintf

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method void js_std_init_handlers(JSRuntime *rt)

```
....
3664.           fprintf(stderr, "Could not allocate memory for the
worker");
```

Improper Resource Access Authorization\Path 15:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=314
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3720	3720
Object	fprintf	fprintf

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static void js_dump_obj(JSContext *ctx, FILE *f, JSValueConst val)

```
....
3720.           fprintf(f, "%s\n", str);
```

Improper Resource Access Authorization\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=315
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c

Line	3723	3723
Object	fprintf	fprintf

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static void js_dump_obj(JSContext *ctx, FILE *f, JSValueConst val)

```
....
3723.           fprintf(f, "[exception]\n");
```

Improper Resource Access Authorization\Path 17:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=316
Status New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3757	3757
Object	fprintf	fprintf

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method void js_std_promise_rejection_tracker(JSContext *ctx, JSValueConst promise,

```
....
3757.           fprintf(stderr, "Possibly unhandled promise rejection:
");
```

Improper Resource Access Authorization\Path 18:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=317
Status New

	Source	Destination
File	AliOS-Things-2/tool_filetime.c	AliOS-Things-2/tool_filetime.c
Line	51	51
Object	fprintf	fprintf

Code Snippet
File Name AliOS-Things-2/tool_filetime.c
Method curl_off_t getfiletime(const char *filename, FILE *error_stream)

```
....
51.          fprintf(error_stream,
```

Improper Resource Access Authorization\Path 19:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=318
Status	New

	Source	Destination
File	AliOS-Things-2/tool_filetime.c	AliOS-Things-2/tool_filetime.c
Line	59	59
Object	fprintf	fprintf

Code Snippet
File Name AliOS-Things-2/tool_filetime.c
Method curl_off_t getfiletime(const char *filename, FILE *error_stream)

```
....
59.         fprintf(error_stream,
```

Improper Resource Access Authorization\Path 20:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=319
Status	New

	Source	Destination
File	AliOS-Things-2/tool_filetime.c	AliOS-Things-2/tool_filetime.c
Line	67	67
Object	fprintf	fprintf

Code Snippet
File Name AliOS-Things-2/tool_filetime.c
Method curl_off_t getfiletime(const char *filename, FILE *error_stream)

```
....
67.       fprintf(error_stream,
```

Improper Resource Access Authorization\Path 21:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=320
Status	New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	580	580
Object	fprintf	fprintf

Code Snippet
File Name AliOS-Things-2/url.c
Method CURLcode Curl_open(struct Curl_easy **curl)

```
....
580.      DEBUGF(fprintf(stderr, "Error: calloc of Curl_easy
failed\n"));
```

Improper Resource Access Authorization\Path 22:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=321
Status	New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	588	588
Object	fprintf	fprintf

Code Snippet
File Name AliOS-Things-2/url.c
Method CURLcode Curl_open(struct Curl_easy **curl)

```
....
588.      DEBUGF(fprintf(stderr, "Error: resolver_init failed\n"));
```

Improper Resource Access Authorization\Path 23:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=322
Status	New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c

Line	597	597
Object	fprintf	fprintf

Code Snippet
File Name AliOS-Things-2/url.c
Method CURLcode Curl_open(struct Curl_easy **curl)

```
....
597.      DEBUGF(fprintf(stderr, "Error: malloc of buffer failed\n"));
```

Improper Resource Access Authorization\Path 24:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=323
Status New

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	603	603
Object	fprintf	fprintf

Code Snippet
File Name AliOS-Things-2/url.c
Method CURLcode Curl_open(struct Curl_easy **curl)

```
....
603.        DEBUGF(fprintf(stderr, "Error: malloc of headerbuff
failed\n"));
```

Improper Resource Access Authorization\Path 25:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=324
Status New

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1532	1532
Object	fputs	fputs

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static int cookie_output(struct CookieInfo *c, const char *dumphere)

```
....
1532.    fputs("# Netscape HTTP Cookie File\n"
```

Improper Resource Access Authorization\Path 26:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=325
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	1191	1191
Object	fputc	fputc

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_std_file_putByte(JSContext *ctx, JSValueConst this_val,

```
....
1191.       c = fputc(c, f);
```

Improper Resource Access Authorization\Path 27:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=326
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	350	350
Object	fwrite	fwrite

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_printf_internal(JSContext *ctx,

```
....
350.                len = fwrite(dbuf.buf, 1, dbuf.size, fp);
```

Improper Resource Access Authorization\Path 28:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=327
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	949	949
Object	fwrite	fwrite

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_std_file_puts(JSContext *ctx, JSValueConst this_val,

```
....
949.            fwrite(str, 1, len, f);
```

Improper Resource Access Authorization\Path 29:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=328
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	1088	1088
Object	fwrite	fwrite

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_std_file_read_write(JSContext *ctx, JSValueConst this_val,

```
....
1088.           ret = fwrite(buf + pos, 1, len, f);
```

Improper Resource Access Authorization\Path 30:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=329
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3621	3621

Object	fwrite	fwrite

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_print(JSContext *ctx, JSValueConst this_val,

```
....
3621.           fwrite(str, 1, len, stdout);
```

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-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=85
Status	New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by hostname_resolve at AliOS-Things-2/url.c in line 3304.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3410	3360
Object	null	hostname_resolve

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name AliOS-Things-2/url.c
Method static void reuse_conn(struct connectdata *old_conn,

```
....
3360.    Curl_safefree(conn->hostname_resolve);
```

NULL Pointer Dereference\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=86
Status	New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by socks_proxy at AliOS-Things-2/url.c in line 3304.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3410	3339
Object	null	socks_proxy

Code Snippet

File Name AliOS-Things-2/url.c

Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name AliOS-Things-2/url.c

Method static void reuse_conn(struct connectdata *old_conn,

```
....
3339.      Curl_safefree(conn->socks_proxy.passwd);
```

NULL Pointer Dereference\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=87
Status	New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by socks_proxy at AliOS-Things-2/url.c in line 3304.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3410	3337
Object	null	socks_proxy

Code Snippet

File Name AliOS-Things-2/url.c

Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name AliOS-Things-2/url.c

Method static void reuse_conn(struct connectdata *old_conn,

```
....
3337.      Curl_safefree(conn->socks_proxy.user);
```

NULL Pointer Dereference\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=88
Status	New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by conn_to_host at AliOS-Things-2/url.c in line 3304.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3410	3355
Object	null	conn_to_host

Code Snippet

File Name AliOS-Things-2/url.c

Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name AliOS-Things-2/url.c

Method static void reuse_conn(struct connectdata *old_conn,

```
....
3355.    Curl_safefree(conn->conn_to_host.rawalloc);
```

NULL Pointer Dereference\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=89
Status	New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by conn_to_host at AliOS-Things-2/url.c in line 3304.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3410	3353
Object	null	conn_to_host

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name AliOS-Things-2/url.c

Method static void reuse_conn(struct connectdata *old_conn,

```
....
3353.    free_idnconverted_hostname(&conn->conn_to_host);
```

NULL Pointer Dereference\Path 6:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=90
Status New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by handler at AliOS-Things-2/url.c in line 851.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3410	857
Object	null	handler

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name AliOS-Things-2/url.c

Method	static int IsMultiplexingPossible(const struct Curl_easy *handle,

```
....
857.    if((conn->handler->protocol & PROTO_FAMILY_HTTP) &&
```

NULL Pointer Dereference\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=91
Status	New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by handler at AliOS-Things-2/url.c in line 3999.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3410	4008
Object	null	handler

Code Snippet

File Name	AliOS-Things-2/url.c
Method	static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name	AliOS-Things-2/url.c
Method	CURLcode Curl_init_do(struct Curl_easy *data, struct connectdata *conn)

```
....
4008.         !(conn->handler->flags & PROTOPT_WILDCARD))
```

NULL Pointer Dereference\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=92
Status	New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by user at AliOS-Things-2/url.c in line 3304.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c

Line	3410	3325
Object	null	user

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name AliOS-Things-2/url.c

Method static void reuse_conn(struct connectdata *old_conn,

```
....
3325.      Curl_safefree(conn->user);
```

NULL Pointer Dereference\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=93
Status	New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by passwd at AliOS-Things-2/url.c in line 3304.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3410	3326
Object	null	passwd

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name AliOS-Things-2/url.c

Method static void reuse_conn(struct connectdata *old_conn,

```
....
3326.      Curl_safefree(conn->passwd);
```

NULL Pointer Dereference\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=94
Status	New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by host at AliOS-Things-2/url.c in line 3304.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3410	3354
Object	null	host

Code Snippet

File Name AliOS-Things-2/url.c

Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name AliOS-Things-2/url.c

Method static void reuse_conn(struct connectdata *old_conn,

```
....
3354.    Curl_safefree(conn->host.rawalloc);
```

NULL Pointer Dereference\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=95
Status	New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by host at AliOS-Things-2/url.c in line 3304.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3410	3352
Object	null	host

Code Snippet

File Name AliOS-Things-2/url.c

Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name AliOS-Things-2/url.c

Method static void reuse_conn(struct connectdata *old_conn,

```
....
3352.    free_idnconverted_hostname(&conn->host);
```

NULL Pointer Dereference\Path 12:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=96
Status	New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by bits at AliOS-Things-2/url.c in line 851.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3410	858
Object	null	bits

Code Snippet

File Name AliOS-Things-2/url.c

Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name AliOS-Things-2/url.c

Method static int IsMultiplexingPossible(const struct Curl_easy *handle,

```
....
858.       (!conn->bits.protoconnstart || !conn->bits.close)) {
```

NULL Pointer Dereference\Path 13:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=97
Status	New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by bits at AliOS-Things-2/url.c in line 851.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3410	858
Object	null	bits

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name AliOS-Things-2/url.c

Method static int IsMultiplexingPossible(const struct Curl_easy *handle,

```
....
858.        (!conn->bits.protoconnstart || !conn->bits.close)) {
```

NULL Pointer Dereference\Path 14:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=98
Status New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by http_proxy at AliOS-Things-2/url.c in line 3304.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3410	3338
Object	null	http_proxy

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name AliOS-Things-2/url.c

Method	static void reuse_conn(struct connectdata *old_conn,

```
....
3338.       Curl_safefree(conn->http_proxy.passwd);
```

NULL Pointer Dereference\Path 15:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=99
Status	New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by http_proxy at AliOS-Things-2/url.c in line 3304.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3410	3336
Object	null	http_proxy

Code Snippet

File Name	AliOS-Things-2/url.c
Method	static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.     struct connectdata *conn_temp = NULL;
```

File Name	AliOS-Things-2/url.c
Method	static void reuse_conn(struct connectdata *old_conn,

```
....
3336.       Curl_safefree(conn->http_proxy.user);
```

NULL Pointer Dereference\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=100
Status	New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by bits at AliOS-Things-2/url.c in line 3304.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c

Line	3410	3334
Object	null	bits

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name AliOS-Things-2/url.c

Method static void reuse_conn(struct connectdata *old_conn,

```
....
3334.    if(conn->bits.proxy_user_passwd) {
```

NULL Pointer Dereference\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=101
Status	New

The variable declared in null at AliOS-Things-2/url.c in line 3404 is not initialized when it is used by bits at AliOS-Things-2/url.c in line 3304.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3410	3323
Object	null	bits

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode create_conn(struct Curl_easy *data,

```
....
3410.    struct connectdata *conn_temp = NULL;
```

File Name AliOS-Things-2/url.c

Method static void reuse_conn(struct connectdata *old_conn,

```
....
3323.    if(conn->bits.user_passwd) {
```

NULL Pointer Dereference\Path 18:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=102
Status	New

The variable declared in 0 at AliOS-Things-2/l2cap.c in line 2101 is not initialized when it is used by rx at AliOS-Things-2/l2cap.c in line 1984.

	Source	Destination
File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	2105	2013
Object	0	rx

Code Snippet

File Name AliOS-Things-2/l2cap.c

Method static void l2cap_chan_le_recv_seg(struct bt_l2cap_le_chan *chan,

```
....
2105.       u16_t seg = 0U;
```

File Name AliOS-Things-2/l2cap.c

Method static void l2cap_chan_send_credits(struct bt_l2cap_le_chan *chan,

```
....
2013.       BT_DBG("chan %p credits %u", chan, atomic_get(&chan-
>rx.credits));
```

NULL Pointer Dereference\Path 19:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=103
Status	New

The variable declared in 0 at AliOS-Things-2/l2cap.c in line 2101 is not initialized when it is used by rx at AliOS-Things-2/l2cap.c in line 874.

	Source	Destination
File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	2105	879
Object	0	rx

Code Snippet

File Name AliOS-Things-2/l2cap.c

Method static void l2cap_chan_le_recv_seg(struct bt_l2cap_le_chan *chan,

```
....
2105.       u16_t seg = 0U;
```

File Name AliOS-Things-2/l2cap.c

Method static void l2cap_chan_rx_give_credits(struct bt_l2cap_le_chan *chan,

```
....
879.        atomic_add(&chan->rx.credits, credits);
```

NULL Pointer Dereference\Path 20:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=104
Status	New

The variable declared in 0 at AliOS-Things-2/l2cap.c in line 2101 is not initialized when it is used by rx at AliOS-Things-2/l2cap.c in line 1984.

	Source	Destination
File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	2105	1991
Object	0	rx

Code Snippet

File Name AliOS-Things-2/l2cap.c

Method static void l2cap_chan_le_recv_seg(struct bt_l2cap_le_chan *chan,

```
....
2105.       u16_t seg = 0U;
```

File Name AliOS-Things-2/l2cap.c

Method static void l2cap_chan_send_credits(struct bt_l2cap_le_chan *chan,

```
....
1991.               credits = chan->rx.init_credits;
```

NULL Pointer Dereference\Path 21:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=105
Status	New

The variable declared in 0 at AliOS-Things-2/l2cap.c in line 2101 is not initialized when it is used by rx at AliOS-Things-2/l2cap.c in line 1984.

	Source	Destination
File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	2105	1990
Object	0	rx

Code Snippet
File Name AliOS-Things-2/l2cap.c
Method static void l2cap_chan_le_recv_seg(struct bt_l2cap_le_chan *chan,

```
....
2105.       u16_t seg = 0U;
```

File Name AliOS-Things-2/l2cap.c
Method static void l2cap_chan_send_credits(struct bt_l2cap_le_chan *chan,

```
....
1990.       if (credits > chan->rx.init_credits) {
```

NULL Pointer Dereference\Path 22:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=106
Status New

The variable declared in ch at AliOS-Things-2/l2cap.c in line 1068 is not initialized when it is used by rx at AliOS-Things-2/l2cap.c in line 1068.

	Source	Destination
File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	1073	1122
Object	ch	rx

Code Snippet
File Name AliOS-Things-2/l2cap.c
Method static void le_ecred_conn_req(struct bt_l2cap *l2cap, u8_t ident,

```
....
1073.       struct bt_l2cap_le_chan *ch = NULL;
....
1122.                   dcid[i++] = sys_cpu_to_le16(ch->rx.cid);
```

NULL Pointer Dereference\Path 23:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=107
Status	New

The variable declared in ch at AliOS-Things-2/l2cap.c in line 1068 is not initialized when it is used by rx at AliOS-Things-2/l2cap.c in line 1068.

	Source	Destination
File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	1073	1158
Object	ch	rx

Code Snippet
File Name AliOS-Things-2/l2cap.c
Method static void le_ecred_conn_req(struct bt_l2cap *l2cap, u8_t ident,

```
....
1073.       struct bt_l2cap_le_chan *ch = NULL;
....
1158.               rsp->mps = sys_cpu_to_le16(ch->rx.mps);
```

NULL Pointer Dereference\Path 24:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=108
Status	New

The variable declared in ch at AliOS-Things-2/l2cap.c in line 1068 is not initialized when it is used by rx at AliOS-Things-2/l2cap.c in line 1068.

	Source	Destination
File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	1073	1160
Object	ch	rx

Code Snippet
File Name AliOS-Things-2/l2cap.c
Method static void le_ecred_conn_req(struct bt_l2cap *l2cap, u8_t ident,

```
....
1073.       struct bt_l2cap_le_chan *ch = NULL;
....
1160.               rsp->credits = sys_cpu_to_le16(ch->rx.init_credits);
```

NULL Pointer Dereference\Path 25:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=109
Status	New

The variable declared in ch at AliOS-Things-2/l2cap.c in line 1068 is not initialized when it is used by rx at AliOS-Things-2/l2cap.c in line 1068.

	Source	Destination
File	AliOS-Things-2/l2cap.c	AliOS-Things-2/l2cap.c
Line	1073	1159
Object	ch	rx

Code Snippet
File Name AliOS-Things-2/l2cap.c
Method static void le_ecred_conn_req(struct bt_l2cap *l2cap, u8_t ident,

```
....
1073.       struct bt_l2cap_le_chan *ch = NULL;
....
1159.               rsp->mtu = sys_cpu_to_le16(ch->rx.mtu);
```

NULL Pointer Dereference\Path 26:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=110
Status	New

The variable declared in nspr_io_stub at AliOS-Things-2/nss.c in line 1802 is not initialized when it is used by secret at AliOS-Things-2/nss.c in line 1802.

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	1806	2000
Object	nspr_io_stub	secret

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_setup_connect(struct connectdata *conn, int sockindex)

```
....
1806.    PRFileDesc *nspr_io_stub = NULL;
....
2000.    nspr_io_stub->secret = (void *)connssl;
```

NULL Pointer Dereference\Path 27:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=111
Status	New

The variable declared in hostaddr at AliOS-Things-2/url.c in line 3181 is not initialized when it is used by addr at AliOS-Things-2/url.c in line 3181.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3202	3216
Object	hostaddr	addr

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode resolve_server(struct Curl_easy *data,

```
....
3202.       struct Curl_dns_entry *hostaddr;
....
3216.           hostaddr->addr = Curl_unix2addr(path, &longpath,
```

NULL Pointer Dereference\Path 28:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=112
Status	New

The variable declared in hostaddr at AliOS-Things-2/url.c in line 3181 is not initialized when it is used by inuse at AliOS-Things-2/url.c in line 3181.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3202	3219
Object	hostaddr	inuse

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode resolve_server(struct Curl_easy *data,

```
....
3202.       struct Curl_dns_entry *hostaddr;
....
3219.             hostaddr->inuse++;
```

NULL Pointer Dereference\Path 29:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=113
Status	New

The variable declared in hostaddr at AliOS-Things-2/url.c in line 3181 is not initialized when it is used by addr at AliOS-Things-2/url.c in line 3181.

	Source	Destination
File	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3202	3218
Object	hostaddr	addr

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode resolve_server(struct Curl_easy *data,

```
....
3202.       struct Curl_dns_entry *hostaddr;
....
3218.           if(hostaddr->addr)
```

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-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=133
Status	New

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	756	756
Object	pathlen	pathlen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method Curl_cookie_add(struct Curl_easy *data,

```
....
756.            co->path[pathlen] = 0; /* zero terminate */
```

Unchecked Array Index\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=134
Status	New

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1084	1084
Object	myhash	myhash

Code Snippet
File Name AliOS-Things-2/cookie.c
Method Curl_cookie_add(struct Curl_easy *data,

```
....
1084.        c->cookies[myhash] = co;
```

Unchecked Array Index\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=135
Status	New

	Source	Destination
File	AliOS-Things-2/escape.c	AliOS-Things-2/escape.c
Line	131	131
Object	strindex	strindex

Code Snippet
File Name AliOS-Things-2/escape.c
Method char *curl_easy_escape(struct Curl_easy *data, const char *string,

```
....
131.    ns[strindex] = 0; /* terminate it */
```

Unchecked Array Index\Path 4:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=136
Status	New

	Source	Destination
File	AliOS-Things-2/escape.c	AliOS-Things-2/escape.c
Line	197	197
Object	strindex	strindex

Code Snippet
File Name AliOS-Things-2/escape.c
Method CURLcode Curl_urldecode(struct Curl_easy *data,

```
....
197.    ns[strindex] = 0; /* terminate it */
```

Unchecked Array Index\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=137
Status	New

	Source	Destination
File	AliOS-Things-2/lossless_msa.c	AliOS-Things-2/lossless_msa.c
Line	196	196
Object	ptemp_dst	ptemp_dst

Code Snippet
File Name AliOS-Things-2/lossless_msa.c
Method int num_pixels, uint8_t* dst) {

```
....
196.      }
```

Unchecked Array Index\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=138
Status	New

	Source	Destination
File	AliOS-Things-2/lossless_msa.c	AliOS-Things-2/lossless_msa.c
Line	196	196

Object	ptemp_dst	ptemp_dst

Code Snippet
File Name AliOS-Things-2/lossless_msa.c
Method int num_pixels, uint8_t* dst) {

```
....
196.      }
```

Unchecked Array Index\Path 7:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=139
Status New

	Source	Destination
File	AliOS-Things-2/lossless_msa.c	AliOS-Things-2/lossless_msa.c
Line	196	196
Object	ptemp_dst	ptemp_dst

Code Snippet
File Name AliOS-Things-2/lossless_msa.c
Method int num_pixels, uint8_t* dst) {

```
....
196.      }
```

Unchecked Array Index\Path 8:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=140
Status New

	Source	Destination
File	AliOS-Things-2/lossless_msa.c	AliOS-Things-2/lossless_msa.c
Line	243	243
Object	ptemp_dst	ptemp_dst

Code Snippet
File Name AliOS-Things-2/lossless_msa.c
Method int num_pixels, uint8_t* dst) {

```
....
243.        }
```

Unchecked Array Index\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=141
Status	New

	Source	Destination
File	AliOS-Things-2/lossless_msa.c	AliOS-Things-2/lossless_msa.c
Line	243	243
Object	ptemp_dst	ptemp_dst

Code Snippet
File Name AliOS-Things-2/lossless_msa.c
Method int num_pixels, uint8_t* dst) {

```
....
243.        }
```

Unchecked Array Index\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=142
Status	New

	Source	Destination
File	AliOS-Things-2/lossless_msa.c	AliOS-Things-2/lossless_msa.c
Line	243	243
Object	ptemp_dst	ptemp_dst

Code Snippet
File Name AliOS-Things-2/lossless_msa.c
Method int num_pixels, uint8_t* dst) {

```
....
243.        }
```

Unchecked Array Index\Path 11:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=143
Status	New

	Source	Destination
File	AliOS-Things-2/mesalink.c	AliOS-Things-2/mesalink.c
Line	263	263
Object	sockindex	sockindex

Code Snippet
File Name AliOS-Things-2/mesalink.c
Method mesalink_connect_step2(struct connectdata *conn, int sockindex)

```
....
263.    conn->recv[sockindex] = mesalink_recv;
```

Unchecked Array Index\Path 12:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=144
Status	New

	Source	Destination
File	AliOS-Things-2/mesalink.c	AliOS-Things-2/mesalink.c
Line	264	264
Object	sockindex	sockindex

Code Snippet
File Name AliOS-Things-2/mesalink.c
Method mesalink_connect_step2(struct connectdata *conn, int sockindex)

```
....
264.    conn->send[sockindex] = mesalink_send;
```

Unchecked Array Index\Path 13:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=145
Status	New

	Source	Destination
File	AliOS-Things-2/mesalink.c	AliOS-Things-2/mesalink.c
Line	555	555

Object	sockindex	sockindex

Code Snippet
File Name AliOS-Things-2/mesalink.c
Method mesalink_connect_common(struct connectdata *conn, int sockindex,

```
....
555.      conn->recv[sockindex] = mesalink_recv;
```

Unchecked Array Index\Path 14:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=146
Status New

	Source	Destination
File	AliOS-Things-2/mesalink.c	AliOS-Things-2/mesalink.c
Line	556	556
Object	sockindex	sockindex

Code Snippet
File Name AliOS-Things-2/mesalink.c
Method mesalink_connect_common(struct connectdata *conn, int sockindex,

```
....
556.      conn->send[sockindex] = mesalink_send;
```

Unchecked Array Index\Path 15:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=147
Status New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	1546	1546
Object	sockindex	sockindex

Code Snippet
File Name AliOS-Things-2/nss.c
Method static void Curl_nss_close(struct connectdata *conn, int sockindex)

```
....
1546.      conn->sock[sockindex] = CURL_SOCKET_BAD;
```

Unchecked Array Index\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=148
Status	New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	2227	2227
Object	sockindex	sockindex

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_connect_common(struct connectdata *conn, int sockindex,

```
....
2227.    conn->recv[sockindex] = nss_recv;
```

Unchecked Array Index\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=149
Status	New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	2228	2228
Object	sockindex	sockindex

Code Snippet
File Name AliOS-Things-2/nss.c
Method static CURLcode nss_connect_common(struct connectdata *conn, int sockindex,

```
....
2228.    conn->send[sockindex] = nss_send;
```

Unchecked Array Index\Path 18:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=150
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	1778	1778
Object	magic	magic

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_os_setReadHandler(JSContext *ctx, JSValueConst this_val,

```
....
1778.              rh->rw_func[magic] = JS_NULL;
```

Unchecked Array Index\Path 19:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=151
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	1799	1799
Object	magic	magic

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_os_setReadHandler(JSContext *ctx, JSValueConst this_val,

```
....
1799.          rh->rw_func[magic] = JS_DupValue(ctx, func);
```

Unchecked Array Index\Path 20:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=152
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	2638	2638

Object	key_len	key_len

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static char **build_envp(JSContext *ctx, JSValueConst obj)

```
....
2638.           pair[key_len] = '=';
```

Unchecked Array Index\Path 21:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=153
Status New

	Source	Destination
File	AliOS-Things-2/SDL_wave.c	AliOS-Things-2/SDL_wave.c
Line	571	571
Object	pos	pos

Code Snippet
File Name AliOS-Things-2/SDL_wave.c
Method MS_ADPCM_DecodeBlockHeader(ADPCM_DecoderState *state)

```
....
571.           state->output.data[state->output.pos] = (Sint16)sample;
```

Unchecked Array Index\Path 22:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=154
Status New

	Source	Destination
File	AliOS-Things-2/SDL_wave.c	AliOS-Things-2/SDL_wave.c
Line	1261	1261
Object	i	i

Code Snippet
File Name AliOS-Things-2/SDL_wave.c
Method LAW_Decode(WaveFile *file, Uint8 **audio_buf, Uint32 *audio_len)

```
....
1261.               dst[i] = alaw_lut[src[i]];
```

Unchecked Array Index\Path 23:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=155
Status	New

	Source	Destination
File	AliOS-Things-2/SDL_wave.c	AliOS-Things-2/SDL_wave.c
Line	1266	1266
Object	i	i

Code Snippet
File Name AliOS-Things-2/SDL_wave.c
Method LAW_Decode(WaveFile *file, Uint8 **audio_buf, Uint32 *audio_len)

```
....
1266.               dst[i] = mulaw_lut[src[i]];
```

Unchecked Array Index\Path 24:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=156
Status	New

	Source	Destination
File	AliOS-Things-2/smtp.c	AliOS-Things-2/smtp.c
Line	867	867
Object	len	len

Code Snippet
File Name AliOS-Things-2/smtp.c
Method static CURLcode smtp_state_command_resp(struct connectdata *conn, int smtpcode,

```
....
867.        line[len] = '\n';
```

Unchecked Array Index\Path 25:

Severity	Low
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=157
Status	New

	Source	Destination
File	AliOS-Things-2/smtp.c	AliOS-Things-2/smtp.c
Line	869	869
Object	len	len

Code Snippet
File Name AliOS-Things-2/smtp.c
Method static CURLcode smtp_state_command_resp(struct connectdata *conn, int smtpcode,

```
....
869.        line[len] = '\0';
```

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-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=7
Status	New

The *nss_sslver_to_name method calls the strdup function, at line 250 of AliOS-Things-2/nss.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	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	254	254
Object	strdup	strdup

Code Snippet
File Name AliOS-Things-2/nss.c
Method static char *nss_sslver_to_name(PRUint16 nssver)

```
....
254.      return strdup("SSLv2");
```

Unchecked Return Value\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=8
Status	New

The *nss_sslver_to_name method calls the strdup function, at line 250 of AliOS-Things-2/nss.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	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	256	256
Object	strdup	strdup

Code Snippet
File Name AliOS-Things-2/nss.c
Method static char *nss_sslver_to_name(PRUint16 nssver)

```
....
256.      return strdup("SSLv3");
```

Unchecked Return Value\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=9
Status	New

The *nss_sslver_to_name method calls the strdup function, at line 250 of AliOS-Things-2/nss.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	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	258	258
Object	strdup	strdup

Code Snippet
File Name AliOS-Things-2/nss.c
Method static char *nss_sslver_to_name(PRUint16 nssver)

```
....
258.      return strdup("TLSv1.0");
```

Unchecked Return Value\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=10
Status	New

The *nss_sslver_to_name method calls the strdup function, at line 250 of AliOS-Things-2/nss.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	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	261	261
Object	strdup	strdup

Code Snippet
File Name AliOS-Things-2/nss.c
Method static char *nss_sslver_to_name(PRUint16 nssver)

```
....
261.      return strdup("TLSv1.1");
```

Unchecked Return Value\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=11
Status	New

The *nss_sslver_to_name method calls the strdup function, at line 250 of AliOS-Things-2/nss.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	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	265	265
Object	strdup	strdup

Code Snippet
File Name AliOS-Things-2/nss.c
Method static char *nss_sslver_to_name(PRUint16 nssver)

```
....
265.      return strdup("TLSv1.2");
```

Unchecked Return Value\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=12
Status	New

The *nss_sslver_to_name method calls the strdup function, at line 250 of AliOS-Things-2/nss.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	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	269	269
Object	strdup	strdup

Code Snippet
File Name AliOS-Things-2/nss.c
Method static char *nss_sslver_to_name(PRUint16 nssver)

```
....
269.      return strdup("TLSv1.3");
```

Unchecked Return Value\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=13
Status	New

The *dup_nickname method calls the strdup function, at line 393 of AliOS-Things-2/nss.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	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	399	399
Object	strdup	strdup

Code Snippet
File Name AliOS-Things-2/nss.c
Method static char *dup_nickname(struct Curl_easy *data, const char *str)

```
....
399.      return strdup(str);
```

Unchecked Return Value\Path 8:

Severity	Low

Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=14
Status	New

The *dup_nickname method calls the strdup function, at line 393 of AliOS-Things-2/nss.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	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	406	406
Object	strdup	strdup

Code Snippet
File Name AliOS-Things-2/nss.c
Method static char *dup_nickname(struct Curl_easy *data, const char *str)

```
....
406.      return strdup(str);
```

Unchecked Return Value\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=15
Status	New

The strstore method calls the Pointer function, at line 369 of AliOS-Things-2/cookie.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	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	372	372
Object	Pointer	Pointer

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static void strstore(char **str, const char *newstr)

```
....
372.    *str = strdup(newstr);
```

Unchecked Return Value\Path 10:

Severity	Low
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=16
Status	New

The krb5_encode method calls the Pointer function, at line 113 of AliOS-Things-2/krb5.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	AliOS-Things-2/krb5.c	AliOS-Things-2/krb5.c
Line	136	136
Object	Pointer	Pointer

Code Snippet
File Name AliOS-Things-2/krb5.c
Method krb5_encode(void *app_data, const void *from, int length, int level, void **to)

```
....
136.     *to = malloc(enc.length);
```

Unchecked Return Value\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=17
Status	New

The js_os_remove method calls the ret function, at line 1699 of AliOS-Things-2/quickjs-libc.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	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	1708	1708
Object	ret	ret

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_os_remove(JSContext *ctx, JSValueConst this_val,

```
....
1708.       ret = js_get_errno(remove(filename));
```

Unchecked Return Value\Path 12:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=18
Status	New

The smtp_init method calls the protop function, at line 1108 of AliOS-Things-2/smtp.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	AliOS-Things-2/smtp.c	AliOS-Things-2/smtp.c
Line	1114	1114
Object	protop	protop

Code Snippet
File Name AliOS-Things-2/smtp.c
Method static CURLcode smtp_init(struct connectdata *conn)

```
....
1114.    smtp = data->req.protop = calloc(sizeof(struct SMTP), 1);
```

Unchecked Return Value\Path 13:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=19
Status	New

The override_login method calls the Pointer function, at line 2795 of AliOS-Things-2/url.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	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2818	2818
Object	Pointer	Pointer

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode override_login(struct Curl_easy *data,

```
....
2818.      *userp = strdup(data->set.str[STRING_USERNAME]);
```

Unchecked Return Value\Path 14:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=600

	84&pathid=20
Status	New

The override_login method calls the Pointer function, at line 2795 of AliOS-Things-2/url.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	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2827	2827
Object	Pointer	Pointer

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode override_login(struct Curl_easy *data,

```
....
2827.       *passwdp = strdup(data->set.str[STRING_PASSWORD]);
```

Unchecked Return Value\Path 15:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=21
Status	New

The override_login method calls the Pointer function, at line 2795 of AliOS-Things-2/url.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	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	2836	2836
Object	Pointer	Pointer

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode override_login(struct Curl_easy *data,

```
....
2836.       *optionsp = strdup(data->set.str[STRING_OPTIONS]);
```

Unchecked Return Value\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=22

Status	New

The parse_connect_to_host_port method calls the Pointer function, at line 2935 of AliOS-Things-2/url.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	AliOS-Things-2/url.c	AliOS-Things-2/url.c
Line	3019	3019
Object	Pointer	Pointer

Code Snippet
File Name AliOS-Things-2/url.c
Method static CURLcode parse_connect_to_host_port(struct Curl_easy *data,

```
....
3019.       *hostname_result = strdup(hostptr);
```

Sizeof Pointer Argument

Query Path:
CPP\Cx\CPP Low Visibility\Sizeof Pointer Argument Version:0

Description

Sizeof Pointer Argument\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=120
Status	New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	299	299
Object	cipherlist	sizeof

Code Snippet
File Name AliOS-Things-2/nss.c
Method static SECStatus set_ciphers(struct Curl_easy *data, PRFileDesc * model,

```
....
299.     for(i = 0; i < NUM_OF_CIPHERS; i++) {
```

Sizeof Pointer Argument\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=121
Status	New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	299	299
Object	cipherlist	sizeof

Code Snippet
File Name AliOS-Things-2/nss.c
Method static SECStatus set_ciphers(struct Curl_easy *data, PRFileDesc * model,

```
....
299.    for(i = 0; i < NUM_OF_CIPHERS; i++) {
```

Sizeof Pointer Argument\Path 3:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=122
Status New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	335	335
Object	cipherlist	sizeof

Code Snippet
File Name AliOS-Things-2/nss.c
Method static SECStatus set_ciphers(struct Curl_easy *data, PRFileDesc * model,

```
....
335.    for(i = 0; i<NUM_OF_CIPHERS; i++) {
```

Sizeof Pointer Argument\Path 4:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=123
Status New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	299	335
Object	cipherlist	sizeof

Code Snippet
File Name AliOS-Things-2/nss.c
Method static SECStatus set_ciphers(struct Curl_easy *data, PRFileDesc * model,

```
....
299.    for(i = 0; i < NUM_OF_CIPHERS; i++) {
....
335.    for(i = 0; i<NUM_OF_CIPHERS; i++) {
```

Sizeof Pointer Argument\Path 5:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=124
Status New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	316	335
Object	cipherlist	sizeof

Code Snippet
File Name AliOS-Things-2/nss.c
Method static SECStatus set_ciphers(struct Curl_easy *data, PRFileDesc * model,

```
....
316.      for(i = 0; i<NUM_OF_CIPHERS; i++) {
....
335.    for(i = 0; i<NUM_OF_CIPHERS; i++) {
```

Sizeof Pointer Argument\Path 6:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=125
Status New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	335	335
Object	cipherlist	sizeof

Code Snippet
File Name AliOS-Things-2/nss.c
Method static SECStatus set_ciphers(struct Curl_easy *data, PRFileDesc * model,

```
....
335.    for(i = 0; i<NUM_OF_CIPHERS; i++) {
```

Sizeof Pointer Argument\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=126
Status	New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	299	335
Object	cipherlist	sizeof

Code Snippet
File Name AliOS-Things-2/nss.c
Method static SECStatus set_ciphers(struct Curl_easy *data, PRFileDesc * model,

```
....
299.    for(i = 0; i < NUM_OF_CIPHERS; i++) {
....
335.    for(i = 0; i<NUM_OF_CIPHERS; i++) {
```

Sizeof Pointer Argument\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=127
Status	New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	316	335
Object	cipherlist	sizeof

Code Snippet
File Name AliOS-Things-2/nss.c
Method static SECStatus set_ciphers(struct Curl_easy *data, PRFileDesc * model,

```
....
316.      for(i = 0; i<NUM_OF_CIPHERS; i++) {
....
335.    for(i = 0; i<NUM_OF_CIPHERS; i++) {
```

Sizeof Pointer Argument\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=128
Status	New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	316	316
Object	cipherlist	sizeof

Code Snippet
File Name AliOS-Things-2/nss.c
Method static SECStatus set_ciphers(struct Curl_easy *data, PRFileDesc * model,

```
....
316.      for(i = 0; i<NUM_OF_CIPHERS; i++) {
```

Sizeof Pointer Argument\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=129
Status	New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	299	316
Object	cipherlist	sizeof

Code Snippet
File Name AliOS-Things-2/nss.c
Method static SECStatus set_ciphers(struct Curl_easy *data, PRFileDesc * model,

```
....
299.    for(i = 0; i < NUM_OF_CIPHERS; i++) {
....
316.      for(i = 0; i<NUM_OF_CIPHERS; i++) {
```

Sizeof Pointer Argument\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=130
Status	New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	316	316
Object	cipherlist	sizeof

Code Snippet
File Name AliOS-Things-2/nss.c
Method static SECStatus set_ciphers(struct Curl_easy *data, PRFileDesc * model,

```
....
316.        for(i = 0; i<NUM_OF_CIPHERS; i++) {
```

Sizeof Pointer Argument\Path 12:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=131
Status New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	299	316
Object	cipherlist	sizeof

Code Snippet
File Name AliOS-Things-2/nss.c
Method static SECStatus set_ciphers(struct Curl_easy *data, PRFileDesc * model,

```
....
299.      for(i = 0; i < NUM_OF_CIPHERS; i++) {
....
316.        for(i = 0; i<NUM_OF_CIPHERS; i++) {
```

Sizeof Pointer Argument\Path 13:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=132
Status New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	2589	2589
Object	buf	sizeof

Code Snippet	
File Name	AliOS-Things-2/quickjs-libc.c
Method	static JSValue js_os_readlink(JSContext *ctx, JSValueConst this_val,

```
....
2589.       res = readlink(path, buf, sizeof(buf) - 1);
```

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-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=330
Status	New

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1526	1526
Object	out	out

Code Snippet	
File Name	AliOS-Things-2/cookie.c
Method	static int cookie_output(struct CookieInfo *c, const char *dumphere)

```
....
1526.       out = fopen(dumphere, FOPEN_WRITETEXT);
```

Incorrect Permission Assignment For Critical Resources\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=331
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	371	371

Object	f	f

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method uint8_t *js_load_file(JSContext *ctx, size_t *pbuf_len, const char *filename)

```
....
371.      f = fopen(filename, "rb");
```

Incorrect Permission Assignment For Critical Resources\Path 3:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=332
Status New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	806	806
Object	f	f

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_std_open(JSContext *ctx, JSValueConst this_val,

```
....
806.      f = fopen(filename, mode);
```

Incorrect Permission Assignment For Critical Resources\Path 4:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=333
Status New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	2340	2340
Object	mkdir	mkdir

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_os_mkdir(JSContext *ctx, JSValueConst this_val,

```
....
2340.       ret = js_get_errno(mkdir(path));
```

Incorrect Permission Assignment For Critical Resources\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=334
Status	New

	Source	Destination
File	AliOS-Things-2/tool_filetime.c	AliOS-Things-2/tool_filetime.c
Line	106	106
Object	CreateFileA	CreateFileA

Code Snippet
File Name AliOS-Things-2/tool_filetime.c
Method void setfiletime(curl_off_t filetime, const char *filename,

```
....
106.      hfile = CreateFileA(filename, FILE_WRITE_ATTRIBUTES,
```

Incorrect Permission Assignment For Critical Resources\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=335
Status	New

	Source	Destination
File	AliOS-Things-2/tool_filetime.c	AliOS-Things-2/tool_filetime.c
Line	40	40
Object	CreateFileA	CreateFileA

Code Snippet
File Name AliOS-Things-2/tool_filetime.c
Method curl_off_t getfiletime(const char *filename, FILE *error_stream)

```
....
40.    hfile = CreateFileA(filename, FILE_READ_ATTRIBUTES,
```

TOCTOU

Query Path:
CPP\Cx\CPP Low Visibility\TOCTOU Version:1

Description

TOCTOU\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=337
Status	New

The *Curl_cookie_init method in AliOS-Things-2/cookie.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1137	1137
Object	fopen	fopen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method struct CookieInfo *Curl_cookie_init(struct Curl_easy *data,

```
....
1137.       fp = file?fopen(file, FOPEN_READTEXT):NULL;
```

TOCTOU\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=338
Status	New

The cookie_output method in AliOS-Things-2/cookie.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1526	1526
Object	fopen	fopen

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static int cookie_output(struct CookieInfo *c, const char *dumphere)

```
....
1526.       out = fopen(dumphere, FOPEN_WRITETEXT);
```

TOCTOU\Path 3:

Severity	Low

Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=339
Status	New

The *js_load_file method in AliOS-Things-2/quickjs-libc.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	371	371
Object	fopen	fopen

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method uint8_t *js_load_file(JSContext *ctx, size_t *pbuf_len, const char *filename)

```
....
371.      f = fopen(filename, "rb");
```

TOCTOU\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=340
Status	New

The js_std_open method in AliOS-Things-2/quickjs-libc.c file utilizes fopen that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	806	806
Object	fopen	fopen

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_std_open(JSContext *ctx, JSValueConst this_val,

```
....
806.      f = fopen(filename, mode);
```

TOCTOU\Path 5:

Severity	Low
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=341
Status	New

The js_os_open method in AliOS-Things-2/quickjs-libc.c file utilizes open that is accessed by other concurrent functionality in a way that is not thread-safe, which may result in a Race Condition over this resource.

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	1513	1513
Object	open	open

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_os_open(JSContext *ctx, JSValueConst this_val,

```
....
1513.       ret = js_get_errno(open(filename, flags, mode));
```

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-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=23
Status	New

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1339	1339
Object	sizeof	sizeof

Code Snippet
File Name AliOS-Things-2/cookie.c
Method struct Cookie *Curl_cookie_getlist(struct CookieInfo *c,

```
....
1339.       array = malloc(sizeof(struct Cookie *) * matches);
```

Use of Sizeof On a Pointer Type\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=600

	84&pathid=24
Status	New

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1349	1349
Object	sizeof	sizeof

Code Snippet
File Name AliOS-Things-2/cookie.c
Method struct Cookie *Curl_cookie_getlist(struct CookieInfo *c,

```
....
1349.       qsort(array, matches, sizeof(struct Cookie *), cookie_sort);
```

Use of Sizeof On a Pointer Type\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=25
Status	New

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1542	1542
Object	sizeof	sizeof

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static int cookie_output(struct CookieInfo *c, const char *dumphere)

```
....
1542.       array = malloc(sizeof(struct Cookie *) * c->numcookies);
```

Use of Sizeof On a Pointer Type\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=26
Status	New

	Source	Destination
File	AliOS-Things-2/cookie.c	AliOS-Things-2/cookie.c
Line	1558	1558

Object	sizeof	sizeof

Code Snippet
File Name AliOS-Things-2/cookie.c
Method static int cookie_output(struct CookieInfo *c, const char *dumphere)

```
....
1558.      qsort(array, c->numcookies, sizeof(struct Cookie *),
cookie_sort_ct);
```

Unreleased Resource Leak

Query Path:
CPP\Cx\CPP Low Visibility\Unreleased Resource Leak Version:0

Categories

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

Description
Unreleased Resource Leak\Path 1:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=48
Status New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	3095	3095
Object	ps	ps

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSWorkerMessagePipe *js_new_message_pipe(void)

```
....
3095.      pthread_mutex_init(&ps->mutex, NULL);
```

Insecure Temporary File

Query Path:
CPP\Cx\CPP Low Visibility\Insecure Temporary File Version:0

Categories

NIST SP 800-53: SC-4 Information in Shared Resources (P1)
OWASP Top 10 2017: A3-Sensitive Data Exposure

Description
Insecure Temporary File\Path 1:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=119
Status	New

	Source	Destination
File	AliOS-Things-2/quickjs-libc.c	AliOS-Things-2/quickjs-libc.c
Line	897	897
Object	tmpfile	tmpfile

Code Snippet
File Name AliOS-Things-2/quickjs-libc.c
Method static JSValue js_std_tmpfile(JSContext *ctx, JSValueConst this_val,

```
....
897.        f = tmpfile();
```

Exposure of System Data to Unauthorized Control Sphere

Query Path:
CPP\Cx\CPP Low Visibility\Exposure of System Data to Unauthorized Control Sphere Version:1

Categories

FISMA 2014: Configuration Management
NIST SP 800-53: AC-3 Access Enforcement (P1)

Description

Exposure of System Data to Unauthorized Control Sphere\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=336
Status	New

The system data read by krb5_auth in the file AliOS-Things-2/krb5.c at line 146 is potentially exposed by krb5_auth found in AliOS-Things-2/krb5.c at line 146.

	Source	Destination
File	AliOS-Things-2/krb5.c	AliOS-Things-2/krb5.c
Line	171	171
Object	perror	perror

Code Snippet
File Name AliOS-Things-2/krb5.c
Method krb5_auth(void *app_data, struct connectdata *conn)

```
....
171.        perror("getsockname()");
```

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-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1060094&projectid=60084&pathid=342
Status	New

	Source	Destination
File	AliOS-Things-2/nss.c	AliOS-Things-2/nss.c
Line	349	349
Object	cipher-	cipher-

Code Snippet
File Name AliOS-Things-2/nss.c
Method * Return true if at least one cipher-suite is enabled. Used to determine

```
....
349.   * Return true if at least one cipher-suite is enabled. Used to
determine
```

Buffer Overflow OutOfBound

Risk

What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

Cause

How does it happen

Buffer Overflows can manifest in numerous different variations. In it's most basic form, the attack controls a buffer, which is then copied to a smaller buffer without size verification. Because the attacker's source buffer is larger than the program's target buffer, the attacker's data overwrites whatever is next on the stack, allowing the attacker to control program structures.
Alternatively, the vulnerability could be the result of improper bounds checking; exposing internal memory addresses outside of their valid scope; allowing the attacker to control the size of the target buffer; or various other forms.

General Recommendations

How to avoid it

- Always perform proper bounds checking before copying buffers or strings.
- Prefer to use safer functions and structures, e.g. safe string classes over `char*`, `strncpy` over `strcpy`, and so on.
- Consistently apply tests for the size of buffers.
- Do not return variable addresses outside the scope of their variables.

Source Code Examples

CPP

Overflowing Buffers

```
const int BUFFER_SIZE = 10;
char buffer[BUFFER_SIZE];

void copyStringToBuffer(char* inputString)
{
    strcpy(buffer, inputString);
}
```

Checked Buffers

```
const int BUFFER_SIZE = 10;
const int MAX_INPUT_SIZE = 256;
char buffer[BUFFER_SIZE];

void copyStringToBuffer(char* inputString)
{
    if (strnlen(inputString, MAX_INPUT_SIZE) < sizeof(buffer))
    {
        strncpy(buffer, inputString, sizeof(buffer));
    }
}
```

Buffer Overflow boundedcpy

Risk

What might happen

Allowing tainted inputs to set the size of how many bytes to copy from source to destination may cause memory corruption, unexpected behavior, instability and data leakage. In some cases, such as when additional and specific areas of memory are also controlled by user input, it may result in code execution.

Cause

How does it happen

Should the size of the amount of bytes to copy from source to destination be greater than the size of the destination, an overflow will occur, and memory beyond the intended buffer will get overwritten. Since this size value is derived from user input, the user may provide an invalid and dangerous buffer size.

General Recommendations

How to avoid it

- Do not trust memory allocation sizes provided by the user; derive them from the copied values instead.
- If memory allocation by a provided value is absolutely required, restrict this size to safe values only. Specifically ensure that this value does not exceed the destination buffer's size.

Source Code Examples

CPP

Size Parameter is Influenced by User Input

```
char dest_buf[10];
memset(dest_buf, '\0', sizeof(dest_buf));
strncpy(dest_buf, src_buf, size); //Assuming size is provided by user input
```

Validating Destination Buffer Length

```
char dest_buf[10];
memset(dest_buf, '\0', sizeof(dest_buf));
if (size < sizeof(dest_buf) && sizeof(src_buf) >= size) //Assuming size is provided by user
input
{
      strncpy(dest_buf, src_buf, size);
}
else
{
      //...
}
```

Buffer Overflow IndexFromInput

Risk

What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

Cause

How does it happen

Buffer Overflows can manifest in numerous different variations. In it's most basic form, the attack controls a buffer, which is then copied to a smaller buffer without size verification. Because the attacker's source buffer is larger than the program's target buffer, the attacker's data overwrites whatever is next on the stack, allowing the attacker to control program structures.
Alternatively, the vulnerability could be the result of improper bounds checking; exposing internal memory addresses outside of their valid scope; allowing the attacker to control the size of the target buffer; or various other forms.

General Recommendations

How to avoid it

- Always perform proper bounds checking before copying buffers or strings.
- Prefer to use safer functions and structures, e.g. safe string classes over `char*`, `strncpy` over `strcpy`, and so on.
- Consistently apply tests for the size of buffers.
- Do not return variable addresses outside the scope of their variables.

Source Code Examples

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

```
public float getAverage(HttpServletRequest req) {
      int total = Integer.parseInt(req.getParameter("total"));
      int count = Integer.parseInt(req.getParameter("count"));

      return total / count;
}
```

Checked Division

```
public float getAverage(HttpServletRequest req) {
      int total = Integer.parseInt(req.getParameter("total"));
      int count = Integer.parseInt(req.getParameter("count"));
```

```
	if (count > 0)
		return total / count;
	else
		return 0;
}
```

Buffer Overflow boundcpy WrongSizeParam

Risk

What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

Cause

How does it happen

Buffer Overflows can manifest in numerous different variations. In it's most basic form, the attack controls a buffer, which is then copied to a smaller buffer without size verification. Because the attacker's source buffer is larger than the program's target buffer, the attacker's data overwrites whatever is next on the stack, allowing the attacker to control program structures.
Alternatively, the vulnerability could be the result of improper bounds checking; exposing internal memory addresses outside of their valid scope; allowing the attacker to control the size of the target buffer; or various other forms.

General Recommendations

How to avoid it

- Always perform proper bounds checking before copying buffers or strings.
- Prefer to use safer functions and structures, e.g. safe string classes over `char*`, `strncpy` over `strcpy`, and so on.
- Consistently apply tests for the size of buffers.
- Do not return variable addresses outside the scope of their variables.

Source Code Examples

MemoryFree on StackVariable

Risk

What might happen

Undefined Behavior may result with a crash. Crashes may give an attacker valuable information about the system and the program internals. Furthermore, it may leave unprotected files (e.g memory) that may be exploited.

Cause

How does it happen

Calling free() on a variable that was not dynamically allocated (e.g. malloc) will result with an Undefined Behavior.

General Recommendations

How to avoid it

Use free() only on dynamically allocated variables in order to prevent unexpected behavior from the compiler.

Source Code Examples

CPP

Bad - Calling free() on a static variable

```
void clean_up(){
  char temp[256];
  do_something();
  free(tmp);
  return;
}
```

Good - Calling free() only on variables that were dynamically allocated

```
void clean_up(){
  char *buff;
  buff = (char*) malloc(1024);
  free(buff);
  return;
}
```

Wrong Size t Allocation

Risk

What might happen

Incorrect allocation of memory may result in unexpected behavior by either overwriting sections of memory with unexpected values. Under certain conditions where both an incorrect allocation of memory and the values being written can be controlled by an attacker, such an issue may result in execution of malicious code.

Cause

How does it happen

Some memory allocation functions require a size value to be provided as a parameter. The allocated size should be derived from the provided value, by providing the length value of the intended source, multiplied by the size of that length. Failure to perform the correct arithmetic to obtain the exact size of the value will likely result in the source overflowing its destination.

General Recommendations

How to avoid it

- Always perform the correct arithmetic to determine size.
- Specifically for memory allocation, calculate the allocation size from the allocation source:
  - Derive the size value from the length of intended source to determine the amount of units to be processed.
  - Always programmatically consider the size of the each unit and their conversion to memory units - for example, by using sizeof() on the unit's type.
  - Memory allocation should be a multiplication of the amount of units being written, times the size of each unit.

Source Code Examples

CPP

Allocating and Assigning Memory without Sizeof Arithmetic

```
int *ptr;
ptr = (int*)malloc(5);
for (int i = 0; i < 5; i++)
{
      ptr[i] = i * 2 + 1;
}
```

Allocating and Assigning Memory with Sizeof Arithmetic

```
int *ptr;
ptr = (int*)malloc(5 * sizeof(int));
```

```
for (int i = 0; i < 5; i++)
{
      ptr[i] = i * 2 + 1;
}
```

Incorrect Arithmetic of Multi-Byte String Allocation

```
wchar_t * dest;
dest = (wchar_t *)malloc(wcslen(source) + 1); // Would not crash for a short "source"
wcscpy((wchar_t *)dest, source);
wprintf(L"Dest: %s\r\n", dest);
```

Correct Arithmetic of Multi-Byte String Allocation

```
wchar_t * dest;
dest = (wchar_t *)malloc((wcslen(source) + 1) * sizeof(wchar_t));
wcscpy((wchar_t *)dest, source);
wprintf(L"Dest: %s\r\n", dest);
```

Boolean 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

Integer Overflow

Risk

What might happen

Assigning large data types into smaller data types, without proper checks and explicit casting, will lead to undefined behavior and unintentional effects, such as data corruption (e.g. value wraparound, wherein maximum values become minimum values); system crashes; infinite loops; logic errors, such as bypassing of security mechanisms; or even buffer overflows leading to arbitrary code execution.

Cause

How does it happen

This flaw can occur when implicitly casting numerical data types of a larger size, into a variable with a data type of a smaller size. This forces the program to discard some bits of information from the number. Depending on how the numerical data types are stored in memory, this is often the bits with the highest value, causing substantial corruption of the stored number. Alternatively, the sign bit of a signed integer could be lost, completely reversing the intention of the number.

General Recommendations

How to avoid it

- Avoid casting larger data types to smaller types.
- Prefer promoting the target variable to a large enough data type.
- If downcasting is necessary, always check that values are valid and in range of the target type, before casting

---

Source Code Examples

CPP

Unsafe Downsize Casting

```
int unsafe_addition(short op1, int op2) {

    // op2 gets forced from int into a short
    short total = op1 + op2;

    return total;
}
```

Safer Use of Proper Data Types

```
int safe_addition(short op1, int op2) {

    // total variable is of type int, the largest type that is needed
    int total = 0;

    // check if total will overflow available integer size
    if (INT_MAX - abs(op2) > op1)
```

```
    {
        total = op1 + op2;
    }
    else
    {
        // instead of overflow, saturate (but this is not always a good thing)
        total = INT_MAX
    }

    return total;
}
```

Dangerous Functions

Risk

What might happen

Use of dangerous functions may expose varying risks associated with each particular function, with potential impact of improper usage of these functions varying significantly. The presence of such functions indicates a flaw in code maintenance policies and adherence to secure coding practices, in a way that has allowed introducing known dangerous code into the application.

Cause

How does it happen

A dangerous function has been identified within the code. Functions are often deemed dangerous to use for numerous reasons, as there are different sets of vulnerabilities associated with usage of such functions. For example, some string copy and concatenation functions are vulnerable to Buffer Overflow, Memory Disclosure, Denial of Service and more. Use of these functions is not recommended.

General Recommendations

How to avoid it

- Deploy a secure and recommended alternative to any functions that were identified as dangerous.
    - If no secure alternative is found, conduct further researching and testing to identify whether current usage successfully sanitizes and verifies values, and thus successfully avoids the use-cases for whom the function is indeed dangerous
- Conduct a periodical review of methods that are in use, to ensure that all external libraries and built-in functions are up-to-date and whose use has not been excluded from best secure coding practices.

Source Code Examples

CPP

Buffer Overflow in gets()

```
int main()
{
	char buf[10];

	printf("Please enter your name: ");
	gets(buf); // veryveryverylongname
	if (buf == ACCEPTED_NAME)
	{
		// Do something
	}
	return 0;
}
```

Safe reading from user

```
int main()
{
	char buf[10];

	printf("Please enter your name: ");
	fgets(buf, sizeof(buf), stdin); //setting the amount of bytes to read
	if (buf == ACCEPTED_NAME)
	{
		//Do something
	}
	return 0;
}
```

Unsafe function for string copy

```
int main(int argc, char* argv[])
{
	char buf[10];
	strcpy(buf, argv[1]); // overflow occurs when len(argv[1]) > 10 bytes

	return 0;
}
```

Safe string copy

```
int main(int argc, char* argv[])
{
	char buf[10];
	strncpy(buf, argv[1], sizeof(buf));
	buf[9]= '\0'; //strncpy doesn't NULL terminates

	return 0;
}
```

Unsafe format string

```
int main(int argc, char* argv[])
{
	printf(argv[1]); // If argv[1] contains a format token, such as %s,%x or %d, will cause
an access violation
	return 0;
}
```

Safe format string

```
int main(int argc, char* argv[])
{
      printf("%s", argv[1]); // Second parameter is not a formattable string
      return 0;
}
```

Double Free

Weakness ID: 415 *(Weakness Variant)* **Status:** Draft

Description

Description Summary

The product calls free() twice on the same memory address, potentially leading to modification of unexpected memory locations.

Extended Description

When a program calls free() twice with the same argument, the program's memory management data structures become corrupted. This corruption can cause the program to crash or, in some circumstances, cause two later calls to malloc() to return the same pointer. If malloc() returns the same value twice and the program later gives the attacker control over the data that is written into this doubly-allocated memory, the program becomes vulnerable to a buffer overflow attack.

Alternate Terms

Double-free

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Languages

C

C++

Common Consequences

Scope	Effect
Access Control	Doubly freeing memory may result in a write-what-where condition, allowing an attacker to execute arbitrary code.

Likelihood of Exploit

Low to Medium

Demonstrative Examples

Example 1

The following code shows a simple example of a double free vulnerability.

(Bad Code)

Example Language: **C**

```
char* ptr = (char*)malloc (SIZE);
...
if (abrt) {
free(ptr);
}
...
free(ptr);
```

Double free vulnerabilities have two common (and sometimes overlapping) causes:

- Error conditions and other exceptional circumstances
- Confusion over which part of the program is responsible for freeing the memory

Although some double free vulnerabilities are not much more complicated than the previous example, most are spread out across hundreds of lines of code or even different files. Programmers seem particularly susceptible to freeing global variables

more than once.

Example 2

While contrived, this code should be exploitable on Linux distributions which do not ship with heap-chunk check summing turned on.

(Bad Code)

Example Language: **C**

```
#include <stdio.h>
#include <unistd.h>
#define BUFSIZE1 512
#define BUFSIZE2 ((BUFSIZE1/2) - 8)

int main(int argc, char **argv) {
char *buf1R1;
char *buf2R1;
char *buf1R2;
buf1R1 = (char *) malloc(BUFSIZE2);
buf2R1 = (char *) malloc(BUFSIZE2);
free(buf1R1);
free(buf2R1);
buf1R2 = (char *) malloc(BUFSIZE1);
strncpy(buf1R2, argv[1], BUFSIZE1-1);
free(buf2R1);
free(buf1R2);
}
```

Observed Examples

Reference	Description
CVE-2004-0642	Double free resultant from certain error conditions.
CVE-2004-0772	Double free resultant from certain error conditions.
CVE-2005-1689	Double free resultant from certain error conditions.
CVE-2003-0545	Double free from invalid ASN.1 encoding.
CVE-2003-1048	Double free from malformed GIF.
CVE-2005-0891	Double free from malformed GIF.
CVE-2002-0059	Double free from malformed compressed data.

Potential Mitigations

Phase: Architecture and Design

Choose a language that provides automatic memory management.

Phase: Implementation

Ensure that each allocation is freed only once. After freeing a chunk, set the pointer to NULL to ensure the pointer cannot be freed again. In complicated error conditions, be sure that clean-up routines respect the state of allocation properly. If the language is object oriented, ensure that object destructors delete each chunk of memory only once.

Phase: Implementation

Use a static analysis tool to find double free instances.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	398	Indicator of Poor Code Quality	**Seven Pernicious Kingdoms (primary)700**
ChildOf	Category	399	Resource Management Errors	**Development Concepts (primary)699**
ChildOf	Category	633	Weaknesses that Affect Memory	**Resource-specific Weaknesses (primary)631**
ChildOf	Weakness Base	666	Operation on Resource in Wrong Phase of	**Research Concepts (primary)1000**

			Lifetime	
ChildOf	Weakness Class	675	Duplicate Operations on Resource	Research Concepts1000
ChildOf	Category	742	CERT C Secure Coding Section 08 - Memory Management (MEM)	**Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734**
PeerOf	Weakness Base	123	Write-what-where Condition	Research Concepts1000
PeerOf	Weakness Base	416	Use After Free	Development Concepts699 Research Concepts1000
MemberOf	View	630	Weaknesses Examined by SAMATE	**Weaknesses Examined by SAMATE (primary)630**
PeerOf	Weakness Base	364	Signal Handler Race Condition	Research Concepts1000

Relationship Notes

This is usually resultant from another weakness, such as an unhandled error or race condition between threads. It could also be primary to weaknesses such as buffer overflows.

Affected Resources

- Memory

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
PLOVER			DFREE - Double-Free Vulnerability
7 Pernicious Kingdoms			Double Free
CLASP			Doubly freeing memory
CERT C Secure Coding	MEM00-C		Allocate and free memory in the same module, at the same level of abstraction
CERT C Secure Coding	MEM01-C		Store a new value in pointers immediately after free()
CERT C Secure Coding	MEM31-C		Free dynamically allocated memory exactly once

White Box Definitions

A weakness where code path has:

1. start statement that relinquishes a dynamically allocated memory resource
2. end statement that relinquishes the dynamically allocated memory resource

Maintenance Notes

It could be argued that Double Free would be most appropriately located as a child of "Use after Free", but "Use" and "Release" are considered to be distinct operations within vulnerability theory, therefore this is more accurately "Release of a Resource after Expiration or Release", which doesn't exist yet.

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	PLOVER		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Potential Mitigations, Time of Introduction		
2008-08-01		KDM Analytics	External
	added/updated white box definitions		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Description, Maintenance Notes, Relationships, Other Notes, Relationship Notes, Taxonomy Mappings		
2008-11-24	CWE Content Team	MITRE	Internal

	updated Relationships, Taxonomy Mappings		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Other Notes		

BACK TO TOP

Heap Inspection

Risk

What might happen

All variables stored by the application in unencrypted memory can potentially be retrieved by an unauthorized user, with privlieged access to the machine. For example, a privileged attacker could attach a debugger to the running process, or retrieve the process's memory from the swapfile or crash dump file.
Once the attacker finds the user passwords in memory, these can be reused to easily impersonate the user to the system.

Cause

How does it happen

String variables are immutable - in other words, once a string variable is assigned, its value cannot be changed or removed. Thus, these strings may remain around in memory, possibly in multiple locations, for an indefinite period of time until the garbage collector happens to remove it. Sensitive data, such as passwords, will remain exposed in memory as plaintext with no control over their lifetime.

General Recommendations

How to avoid it

Generic Guidance:

- Do not store senstiive data, such as passwords or encryption keys, in memory in plaintext, even for a short period of time.
- Prefer to use specialized classes that store encrypted memory.
- Alternatively, store secrets temporarily in mutable data types, such as byte arrays, and then promptly zeroize the memory locations.

Specific Recommendations - Java:

- Instead of storing passwords in immutable strings, prefer to use an encrypted memory object, such as SealedObject.

Specific Recommendations - .NET:

- Instead of storing passwords in immutable strings, prefer to use an encrypted memory object, such as SecureString or ProtectedData.

Source Code Examples

Java

Plaintext Password in Immutable String

```
class Heap_Inspection
{
  private string password;

  void setPassword()
```

```
  {
      password = System.console().readLine("Enter your password: ");
   }
}
```

Password Protected in Memory

```
class Heap_Inspection_Fixed

{
  private SealedObject password;

  void setPassword()

 {
      byte[] sKey = getKeyFromConfig();
      Cipher c = Cipher.getInstance("AES");
      c.init(Cipher.ENCRYPT_MODE, sKey);

      char[] input = System.console().readPassword("Enter your password: ");
      password = new SealedObject(Arrays.asList(input), c);

    //Zero out the possible password, for security.
    Arrays.fill(password, '0');
  }
}
```

CPP

Vulnerable C code

```
/* Vulnerable to heap inspection */

#include <stdio.h>


void somefunc(){
      printf("Yea, I'm just being called for the heap of it..\n");
}

void authfunc(){
        char* password = (char *) malloc(256);
        char ch;
        ssize_t k;
            int i=0;
        while(k = read(0, &ch, 1) > 0)
        {
                if (ch == '\n'){
                        password[i]='\0';
                        break;
                } else{
                        password[i++]=ch;
                        fflush(0);
                }
        }
        printf("Password: %s\n",&password[0]);
}

int main()

{

    printf("Please enter a password:\n");

      authfunc();
      printf("You can now dump memory to find this password!");
      somefunc();
```

```
	gets();

}
```

Safe C code

```
/* Pesumably safe heap */

#include <stdio.h>
#include <string.h>

#define STDIN_FILENO 0

void somefunc(){
        printf("Yea, I'm just being called for the heap of it..\n");
}

void authfunc(){
	char* password = (char*) malloc(256);
	int i=0;
	char ch;
	ssize_t k;
	while(k = read(STDIN_FILENO, &ch, 1) > 0)
	{
		if (ch == '\n'){
			password[i]='\0';
			break;
		} else{
			password[i++]=ch;
			fflush(0);
		}
	}
	i=0;
	memset(password,'\0',256);
}

int main()

{

	printf("Please enter a password:\n");
	authfunc();
	somefunc();
	char ch;
	while(read(STDIN_FILENO, &ch, 1) > 0)
	{
		if (ch == '\n')
			break;
	}
}
```

Failure to Release Memory Before Removing Last Reference ('Memory Leak')

Weakness ID: 401 *(Weakness Base)* **Status:** Draft

Description

Description Summary

The software does not sufficiently track and release allocated memory after it has been used, which slowly consumes remaining memory.

Extended Description

This is often triggered by improper handling of malformed data or unexpectedly interrupted sessions.

Terminology Notes

"memory leak" has sometimes been used to describe other kinds of issues, e.g. for information leaks in which the contents of memory are inadvertently leaked (CVE-2003-0400 is one such example of this terminology conflict).

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Languages

C

C++

Modes of Introduction

Memory leaks have two common and sometimes overlapping causes:

- Error conditions and other exceptional circumstances
- Confusion over which part of the program is responsible for freeing the memory

Common Consequences

Scope	Effect
Availability	Most memory leaks result in general software reliability problems, but if an attacker can intentionally trigger a memory leak, the attacker might be able to launch a denial of service attack (by crashing or hanging the program) or take advantage of other unexpected program behavior resulting from a low memory condition.

Likelihood of Exploit

Medium

Demonstrative Examples

Example 1

The following C function leaks a block of allocated memory if the call to read() fails to return the expected number of bytes:

(Bad Code)

Example Language: **C**

```
char* getBlock(int fd) {
char* buf = (char*) malloc(BLOCK_SIZE);
if (!buf) {
return NULL;
}
if (read(fd, buf, BLOCK_SIZE) != BLOCK_SIZE) {

return NULL;
}
```

```
return buf;
}
```

Example 2

Here the problem is that every time a connection is made, more memory is allocated. So if one just opened up more and more connections, eventually the machine would run out of memory.

(Bad Code)

Example Language: **C**

```
bar connection(){
foo = malloc(1024);
return foo;
}
endConnection(bar foo) {

free(foo);
}
int main() {

while(1) //thread 1
//On a connection
foo=connection(); //thread 2
//When the connection ends
endConnection(foo)
}
```

Observed Examples

Reference	Description
CVE-2005-3119	Memory leak because function does not free() an element of a data structure.
CVE-2004-0427	Memory leak when counter variable is not decremented.
CVE-2002-0574	Memory leak when counter variable is not decremented.
CVE-2005-3181	Kernel uses wrong function to release a data structure, preventing data from being properly tracked by other code.
CVE-2004-0222	Memory leak via unknown manipulations as part of protocol test suite.
CVE-2001-0136	Memory leak via a series of the same command.

Potential Mitigations

Pre-design: Use a language or compiler that performs automatic bounds checking.

Phase: Architecture and Design

Use an abstraction library to abstract away risky APIs. Not a complete solution.

Pre-design through Build: The Boehm-Demers-Weiser Garbage Collector or valgrind can be used to detect leaks in code. This is not a complete solution as it is not 100% effective.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	398	Indicator of Poor Code Quality	**Seven Pernicious Kingdoms (primary)700**
ChildOf	Category	399	Resource Management Errors	**Development Concepts (primary)699**
ChildOf	Category	633	Weaknesses that Affect Memory	**Resource-specific Weaknesses (primary)631**
ChildOf	Category	730	OWASP Top Ten 2004 Category A9 - Denial of Service	**Weaknesses in OWASP Top Ten (2004) (primary)711**
ChildOf	Weakness Base	772	Missing Release of Resource after Effective	**Research Concepts (primary)1000**

			Lifetime	
MemberOf	View	630	Weaknesses Examined by SAMATE	**Weaknesses Examined by SAMATE (primary)630**
CanFollow	Weakness Class	390	Detection of Error Condition Without Action	Research Concepts1000

Relationship Notes

This is often a resultant weakness due to improper handling of malformed data or early termination of sessions.

Affected Resources

- Memory

Functional Areas

- Memory management

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
PLOVER			Memory leak
7 Pernicious Kingdoms			Memory Leak
CLASP			Failure to deallocate data
OWASP Top Ten 2004	A9	CWE More Specific	Denial of Service

White Box Definitions

A weakness where the code path has:

1. start statement that allocates dynamically allocated memory resource

2. end statement that loses identity of the dynamically allocated memory resource creating situation where dynamically allocated memory resource is never relinquished

Where "loses" is defined through the following scenarios:

1. identity of the dynamic allocated memory resource never obtained

2. the statement assigns another value to the data element that stored the identity of the dynamically allocated memory resource and there are no aliases of that data element

3. identity of the dynamic allocated memory resource obtained but never passed on to function for memory resource release

4. the data element that stored the identity of the dynamically allocated resource has reached the end of its scope at the statement and there are no aliases of that data element

References

J. Whittaker and H. Thompson. "How to Break Software Security". Addison Wesley. 2003.

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	PLOVER		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-01		KDM Analytics	External
	added/updated white box definitions		
2008-08-15		Veracode	External
	Suggested OWASP Top Ten 2004 mapping		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Relationships, Other Notes, References, Relationship Notes, Taxonomy Mappings, Terminology Notes		
2008-10-14	CWE Content Team	MITRE	Internal
	updated Description		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Other Notes		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Name		
2009-07-17	KDM Analytics		External
	Improved the White Box Definition		

2009-07-27	CWE Content Team	MITRE	Internal
	updated White Box Definitions		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Modes of Introduction, Other Notes		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Relationships		

Previous Entry Names

Change Date	Previous Entry Name
2008-04-11	Memory Leak
2009-05-27	Failure to Release Memory Before Removing Last Reference (aka 'Memory Leak')

BACK TO TOP

Use of Uninitialized Pointer

Risk

What might happen

A null pointer dereference is likely to cause a run-time exception, a crash, or other unexpected behavior.

Cause

How does it happen

Variables which are declared without being assigned will implicitly retain a null value until they are assigned. The null value can also be explicitly set to a variable, to ensure clear out its contents. Since null is not really a value, it may not have object variables and methods, and any attempt to access contents of a null object, instead of verifying it is set beforehand, will result in a null pointer dereference exception.

General Recommendations

How to avoid it

- For any variable that is created, ensure all logic flows between declaration and use assign a non-null value to the variable first.
- Enforce null checks on any received variable or object before it is dereferenced, to ensure it does not contain a null assigned to it elsewhere.
- Consider the need to assign null values in order to overwrite initialized variables. Consider reassigning or releasing these variables instead.

Source Code Examples

Use of Uninitialized Variable

Weakness ID: 457 *(Weakness Variant)* **Status:** Draft

Description

Description Summary

The code uses a variable that has not been initialized, leading to unpredictable or unintended results.

Extended Description

In some languages, such as C, an uninitialized variable contains contents of previously-used memory. An attacker can sometimes control or read these contents.

Time of Introduction

- Implementation

Applicable Platforms

Languages

C: *(Sometimes)*

C++: *(Sometimes)*

Perl: *(Often)*

All

Common Consequences

Scope	Effect
Availability Integrity	Initial variables usually contain junk, which can not be trusted for consistency. This can lead to denial of service conditions, or modify control flow in unexpected ways. In some cases, an attacker can "pre-initialize" the variable using previous actions, which might enable code execution. This can cause a race condition if a lock variable check passes when it should not.
Authorization	Strings that are not initialized are especially dangerous, since many functions expect a null at the end -- and only at the end -- of a string.

Likelihood of Exploit

High

Demonstrative Examples

Example 1

The following switch statement is intended to set the values of the variables aN and bN, but in the default case, the programmer has accidentally set the value of aN twice. As a result, bN will have an undefined value.

(Bad Code)

Example Language: **C**

```
switch (ctl) {
case -1:
aN = 0;
bN = 0;
break;
case 0:
aN = i;
bN = -i;
break;
case 1:
aN = i + NEXT_SZ;
bN = i - NEXT_SZ;
break;
default:
```

```
aN = -1;
aN = -1;
break;
}
repaint(aN, bN);
```

Most uninitialized variable issues result in general software reliability problems, but if attackers can intentionally trigger the use of an uninitialized variable, they might be able to launch a denial of service attack by crashing the program. Under the right circumstances, an attacker may be able to control the value of an uninitialized variable by affecting the values on the stack prior to the invocation of the function.

Example 2

Example Languages: **C++ and Java**

```
int foo;
void bar() {
if (foo==0)
/.../
/../
}
```

Observed Examples

Reference	Description
CVE-2008-0081	Uninitialized variable leads to code execution in popular desktop application.
CVE-2007-4682	Crafted input triggers dereference of an uninitialized object pointer.
CVE-2007-3468	Crafted audio file triggers crash when an uninitialized variable is used.
CVE-2007-2728	Uninitialized random seed variable used.

Potential Mitigations

Phase: Implementation

Assign all variables to an initial value.

Phase: Build and Compilation

Most compilers will complain about the use of uninitialized variables if warnings are turned on.

Phase: Requirements

The choice could be made to use a language that is not susceptible to these issues.

Phase: Architecture and Design

Mitigating technologies such as safe string libraries and container abstractions could be introduced.

Other Notes

Before variables are initialized, they generally contain junk data of what was left in the memory that the variable takes up. This data is very rarely useful, and it is generally advised to pre-initialize variables or set them to their first values early. If one forgets -- in the C language -- to initialize, for example a char *, many of the simple string libraries may often return incorrect results as they expect the null termination to be at the end of a string.

Stack variables in C and C++ are not initialized by default. Their initial values are determined by whatever happens to be in their location on the stack at the time the function is invoked. Programs should never use the value of an uninitialized variable. It is not uncommon for programmers to use an uninitialized variable in code that handles errors or other rare and exceptional circumstances. Uninitialized variable warnings can sometimes indicate the presence of a typographic error in the code.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	398	Indicator of Poor Code Quality	**Seven Pernicious Kingdoms (primary)700**
ChildOf	Weakness Base	456	Missing Initialization	**Development Concepts (primary)699 Research Concepts**

				(primary)1000
MemberOf	View	630	Weaknesses Examined by SAMATE	**Weaknesses Examined by SAMATE (primary)630**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
CLASP			Uninitialized variable
7 Pernicious Kingdoms			Uninitialized Variable

White Box Definitions

A weakness where the code path has:

1. start statement that defines variable
2. end statement that accesses the variable
3. the code path does not contain a statement that assigns value to the variable

References

mercy. "Exploiting Uninitialized Data". Jan 2006. < http://www.felinemenace.org/~mercy/papers/UBehavior/UBehavior.zip>.

Microsoft Security Vulnerability Research & Defense. "MS08-014 : The Case of the Uninitialized Stack Variable Vulnerability". 2008-03-11. <http://blogs.technet.com/swi/archive/2008/03/11/the-case-of-the-uninitialized-stack-variable-vulnerability.aspx>.

Content History

Submissions

Submission Date	Submitter	Organization	Source
	CLASP		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-01		KDM Analytics	External
	added/updated white box definitions		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Description, Relationships, Observed Example, Other Notes, References, Taxonomy Mappings		
2009-01-12	CWE Content Team	MITRE	Internal
	updated Common Consequences, Demonstrative Examples, Potential Mitigations		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		

Previous Entry Names

Change Date	Previous Entry Name
2008-04-11	Uninitialized Variable

BACK TO TOP

Use of Zero Initialized Pointer

Risk

What might happen

A null pointer dereference is likely to cause a run-time exception, a crash, or other unexpected behavior.

Cause

How does it happen

Variables which are declared without being assigned will implicitly retain a null value until they are assigned. The null value can also be explicitly set to a variable, to ensure clear out its contents. Since null is not really a value, it may not have object variables and methods, and any attempt to access contents of a null object, instead of verifying it is set beforehand, will result in a null pointer dereference exception.

General Recommendations

How to avoid it

- For any variable that is created, ensure all logic flows between declaration and use assign a non-null value to the variable first.
- Enforce null checks on any received variable or object before it is dereferenced, to ensure it does not contain a null assigned to it elsewhere.
- Consider the need to assign null values in order to overwrite initialized variables. Consider reassigning or releasing these variables instead.

Source Code Examples

CPP

Explicit NULL Dereference

```
char * input = NULL;
printf("%s", input);
```

Implicit NULL Dereference

```
char * input;
printf("%s", input);
```

Java

Explicit Null Dereference

```
Object o = null;
out.println(o.getClass());
```

Inadequate Encryption Strength

Risk

What might happen

Using weak or outdated cryptography does not provide sufficient protection for sensitive data. An attacker that gains access to the encrypted data would likely be able to break the encryption, using either cryptanalysis or brute force attacks. Thus, the attacker would be able to steal user passwords and other personal data. This could lead to user impersonation or identity theft.

Cause

How does it happen

The application uses a weak algorithm, that is considered obselete since it is relatively easy to break. These obselete algorithms are vulnerable to several different kinds of attacks, including brute force.

General Recommendations

How to avoid it

Generic Guidance:

- Always use strong, modern algorithms for encryption, hashing, and so on.
- Do not use weak, outdated, or obsolete algorithms.
- Ensure you select the correct cryptographic mechanism according to the specific requirements.
- Passwords should be protected with a dedicated password protection scheme, such as bcrypt, scrypt, PBKDF2, or Argon2.

Specific Recommendations:

- Do not use SHA-1, MD5, or any other weak hash algorithm to protect passwords or personal data. Instead, use a stronger hash such as SHA-256 when a secure hash is required.
- Do not use DES, Triple-DES, RC2, or any other weak encryption algorithm to protect passwords or personal data. Instead, use a stronger encryption algorithm such as AES to protect personal data.
- Do not use weak encryption modes such as ECB, or rely on insecure defaults. Explicitly specify a stronger encryption mode, such as GCM.
- For symmetric encryption, use a key length of at least 256 bits.

Source Code Examples

Java

Weakly Hashed PII

```
string protectSSN(HttpServletRequest req) {
    string socialSecurityNum = req.getParameter("SocialSecurityNo");

    return DigestUtils.md5Hex(socialSecurityNum);
}
```

Stronger Hash for PII

```
string protectSSN(HttpServletRequest req) {
    string socialSecurityNum = req.getParameter("SocialSecurityNo");

    return DigestUtils.sha256Hex(socialSecurityNum);
}
```

Unchecked Return Value

Risk

What might happen

A program that does not check function return values could cause the application to enter an undefined state. This could lead to unexpected behavior and unintended consequences, including inconsistent data, system crashes or other error-based exploits.

Cause

How does it happen

The application calls a system function, but does not receive or check the result of this funciton. These functions often return error codes in the result, or share other status codes with it's caller. The application simply ignores this result value, losing this vital information.

General Recommendations

How to avoid it

- Always check the result of any called function that returns a value, and verify the result is an expected value.

- Ensure the calling function responds to all possible return values.

- Expect runtime errors and handle them gracefully. Explicitly define a mechanism for handling unexpected errors.

Source Code Examples

CPP

Unchecked Memory Allocation

```
buff = (char*) malloc(size);
strncpy(buff, source, size);
```

Safer Memory Allocation

```
buff = (char*) malloc(size+1);
if (buff==NULL) exit(1);

strncpy(buff, source, size);
buff[size] = '\0';
```

Use of sizeof() on a Pointer Type

Weakness ID: 467 *(Weakness Variant)* **Status:** Draft

Description

Description Summary

The code calls sizeof() on a malloced pointer type, which always returns the wordsize/8. This can produce an unexpected result if the programmer intended to determine how much memory has been allocated.

Time of Introduction

- Implementation

Applicable Platforms

Languages

C

C++

Common Consequences

Scope	Effect
Integrity	This error can often cause one to allocate a buffer that is much smaller than what is needed, leading to resultant weaknesses such as buffer overflows.

Likelihood of Exploit

High

Demonstrative Examples

Example 1

Care should be taken to ensure sizeof returns the size of the data structure itself, and not the size of the pointer to the data structure.

In this example, sizeof(foo) returns the size of the pointer.

(Bad Code)

Example Languages: **C and C++**

```
double *foo;
...
foo = (double *)malloc(sizeof(foo));
```

In this example, sizeof(*foo) returns the size of the data structure and not the size of the pointer.

(Good Code)

Example Languages: **C and C++**

```
double *foo;
...
foo = (double *)malloc(sizeof(*foo));
```

Example 2

This example defines a fixed username and password. The AuthenticateUser() function is intended to accept a username and a password from an untrusted user, and check to ensure that it matches the username and password. If the username and password match, AuthenticateUser() is intended to indicate that authentication succeeded.

(Bad Code)

```
/* Ignore CWE-259 (hard-coded password) and CWE-309 (use of password system for authentication) for this example. */

char *username = "admin";
char *pass = "password";

int AuthenticateUser(char *inUser, char *inPass) {
```

```
printf("Sizeof username = %d\n", sizeof(username));
printf("Sizeof pass = %d\n", sizeof(pass));

if (strncmp(username, inUser, sizeof(username))) {
printf("Auth failure of username using sizeof\n");
return(AUTH_FAIL);
}
/* Because of CWE-467, the sizeof returns 4 on many platforms and architectures. */
if (! strncmp(pass, inPass, sizeof(pass))) {
printf("Auth success of password using sizeof\n");
return(AUTH_SUCCESS);
}
else {
printf("Auth fail of password using sizeof\n");
return(AUTH_FAIL);
}
}

int main (int argc, char **argv)
{
int authResult;

if (argc < 3) {
ExitError("Usage: Provide a username and password");
}
authResult = AuthenticateUser(argv[1], argv[2]);
if (authResult != AUTH_SUCCESS) {
ExitError("Authentication failed");
}
else {
DoAuthenticatedTask(argv[1]);
}
}
```

In AuthenticateUser(), because sizeof() is applied to a parameter with an array type, the sizeof() call might return 4 on many modern architectures. As a result, the strncmp() call only checks the first four characters of the input password, resulting in a partial comparison (CWE-187), leading to improper authentication (CWE-287).

Because of the partial comparison, any of these passwords would still cause authentication to succeed for the "admin" user:

(Attack)

```
pass5
passABCDEFGH
passWORD
```

Because only 4 characters are checked, this significantly reduces the search space for an attacker, making brute force attacks more feasible.

The same problem also applies to the username, so values such as "adminXYZ" and "administrator" will succeed for the username.

Potential Mitigations

Phase: Implementation

Use expressions such as "sizeof(*pointer)" instead of "sizeof(pointer)", unless you intend to run sizeof() on a pointer type to gain some platform independence or if you are allocating a variable on the stack.

Other Notes

The use of sizeof() on a pointer can sometimes generate useful information. An obvious case is to find out the wordsize on a platform. More often than not, the appearance of sizeof(pointer) indicates a bug.

Weakness Ordinalities

Ordinality	Description
Primary	*(where the weakness exists independent of other weaknesses)*

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	465	Pointer Issues	**Development Concepts (primary)699**
ChildOf	Weakness Class	682	Incorrect Calculation	**Research Concepts (primary)1000**
ChildOf	Category	737	CERT C Secure Coding Section 03 - Expressions (EXP)	**Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734**
ChildOf	Category	740	CERT C Secure Coding Section 06 - Arrays (ARR)	Weaknesses Addressed by the CERT C Secure Coding Standard734
CanPrecede	Weakness Base	131	Incorrect Calculation of Buffer Size	Research Concepts1000

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
CLASP			Use of sizeof() on a pointer type
CERT C Secure Coding	ARR01-C		Do not apply the sizeof operator to a pointer when taking the size of an array
CERT C Secure Coding	EXP01-C		Do not take the size of a pointer to determine the size of the pointed-to type

White Box Definitions

A weakness where code path has:

1. end statement that passes an identity of a dynamically allocated memory resource to a sizeof operator
2. start statement that allocates the dynamically allocated memory resource

References

Robert Seacord. "EXP01-A. Do not take the sizeof a pointer to determine the size of a type". <https://www.securecoding.cert.org/confluence/display/seccode/EXP01-A.+Do+not+take+the+sizeof+a+pointer+to+determine+the+size+of+a+type>.

Content History

Submissions

Submission Date	Submitter	Organization	Source
	CLASP		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-01		KDM Analytics	External
	added/updated white box definitions		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Relationships, Other Notes, Taxonomy Mappings, Weakness Ordinalities		
2008-11-24	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Relationships		

BACK TO TOP

Resource Locking Problems

Category ID: 411 *(Category)* **Status:** Draft

Description

Description Summary

Weaknesses in this category are related to improper handling of locks that are used to control access to resources.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	399	Resource Management Errors	**Development Concepts (primary)699**
ParentOf	Weakness Base	412	Unrestricted Externally Accessible Lock	Development Concepts699
ParentOf	Weakness Base	413	Insufficient Resource Locking	**Development Concepts (primary)699**
ParentOf	Weakness Base	414	Missing Lock Check	**Development Concepts (primary)699**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
PLOVER			Resource Locking problems

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	PLOVER		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-09-08	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		

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

Insecure Temporary File

Weakness ID: 377 *(Weakness Base)* **Status:** Incomplete

Description

Description Summary

Creating and using insecure temporary files can leave application and system data vulnerable to attack.

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Languages

All

Demonstrative Examples

Example 1

The following code uses a temporary file for storing intermediate data gathered from the network before it is processed.

(Bad Code)

Example Language: **C**

```
if (tmpnam_r(filename)) {

FILE* tmp = fopen(filename,"wb+");
while((recv(sock,recvbuf,DATA_SIZE, 0) > 0)&(amt!=0)) amt = fwrite(recvbuf,1,DATA_SIZE,tmp);
}
...
```

This otherwise unremarkable code is vulnerable to a number of different attacks because it relies on an insecure method for creating temporary files. The vulnerabilities introduced by this function and others are described in the following sections. The most egregious security problems related to temporary file creation have occurred on Unix-based operating systems, but Windows applications have parallel risks. This section includes a discussion of temporary file creation on both Unix and Windows systems. Methods and behaviors can vary between systems, but the fundamental risks introduced by each are reasonably constant.

Other Notes

Applications require temporary files so frequently that many different mechanisms exist for creating them in the C Library and Windows(R) API. Most of these functions are vulnerable to various forms of attacks.

The functions designed to aid in the creation of temporary files can be broken into two groups based whether they simply provide a filename or actually open a new file. - Group 1: "Unique" Filenames: The first group of C Library and WinAPI functions designed to help with the process of creating temporary files do so by generating a unique file name for a new temporary file, which the program is then supposed to open. This group includes C Library functions like tmpnam(), tempnam(), mktemp() and their C++ equivalents prefaced with an _ (underscore) as well as the GetTempFileName() function from the Windows API. This group of functions suffers from an underlying race condition on the filename chosen. Although the functions guarantee that the filename is unique at the time it is selected, there is no mechanism to prevent another process or an attacker from creating a file with the same name after it is selected but before the application attempts to open the file. Beyond the risk of a legitimate collision caused by another call to the same function, there is a high probability that an attacker will be able to create a malicious collision because the filenames generated by these functions are not sufficiently randomized to make them difficult to guess. If a file with the selected name is created, then depending on how the file is opened the existing contents or access permissions of the file may remain intact. If the existing contents of the file are malicious in nature, an attacker may be able to inject dangerous data into the application when it reads data back from the temporary file. If an attacker pre-creates the file with relaxed access permissions, then data stored in the temporary file by the application may be accessed, modified or corrupted by an attacker. On Unix based systems an even more insidious attack is possible if the attacker pre-creates the file as a link to another important file. Then, if the application truncates or writes data to the file, it may unwittingly perform damaging operations for the attacker. This is an especially serious threat if the program operates with elevated permissions. Finally, in the best case the file will be opened with the a call to open() using the O_CREAT and O_EXCL flags or to CreateFile() using the CREATE_NEW attribute, which will fail if the file already exists and therefore prevent the types of attacks described above. However, if an attacker is able to accurately predict a sequence of temporary file names, then the application may be prevented from opening necessary temporary storage causing a denial of service (DoS) attack. This type of attack would not be difficult to mount given the small amount of randomness used in

the selection of the filenames generated by these functions. - Group 2: "Unique" Files: The second group of C Library functions attempts to resolve some of the security problems related to temporary files by not only generating a unique file name, but also opening the file. This group includes C Library functions like tmpfile() and its C++ equivalents prefaced with an _ (underscore), as well as the slightly better-behaved C Library function mkstemp(). The tmpfile() style functions construct a unique filename and open it in the same way that fopen() would if passed the flags "wb+", that is, as a binary file in read/write mode. If the file already exists, tmpfile() will truncate it to size zero, possibly in an attempt to assuage the security concerns mentioned earlier regarding the race condition that exists between the selection of a supposedly unique filename and the subsequent opening of the selected file. However, this behavior clearly does not solve the function's security problems. First, an attacker can pre-create the file with relaxed access-permissions that will likely be retained by the file opened by tmpfile(). Furthermore, on Unix based systems if the attacker pre-creates the file as a link to another important file, the application may use its possibly elevated permissions to truncate that file, thereby doing damage on behalf of the attacker. Finally, if tmpfile() does create a new file, the access permissions applied to that file will vary from one operating system to another, which can leave application data vulnerable even if an attacker is unable to predict the filename to be used in advance. Finally, mkstemp() is a reasonably safe way create temporary files. It will attempt to create and open a unique file based on a filename template provided by the user combined with a series of randomly generated characters. If it is unable to create such a file, it will fail and return -1. On modern systems the file is opened using mode 0600, which means the file will be secure from tampering unless the user explicitly changes its access permissions. However, mkstemp() still suffers from the use of predictable file names and can leave an application vulnerable to denial of service attacks if an attacker causes mkstemp() to fail by predicting and pre-creating the filenames to be used.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	361	Time and State	**Seven Pernicious Kingdoms (primary)700**
ChildOf	Category	376	Temporary File Issues	**Development Concepts (primary)699**
ChildOf	Weakness Class	668	Exposure of Resource to Wrong Sphere	**Research Concepts (primary)1000**
ParentOf	Weakness Base	378	Creation of Temporary File With Insecure Permissions	**Research Concepts (primary)1000**
ParentOf	Weakness Base	379	Creation of Temporary File in Directory with Incorrect Permissions	**Research Concepts (primary)1000**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
7 Pernicious Kingdoms			Insecure Temporary File

References

[REF-11] M. Howard and D. LeBlanc. "Writing Secure Code". Chapter 23, "Creating Temporary Files Securely" Page 682. 2nd Edition. Microsoft. 2002.

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	7 Pernicious Kingdoms		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Relationships, Other Notes, Taxonomy Mappings		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2010-02-16	CWE Content Team	MITRE	Internal
	updated References		

BACK TO TOP

Use of sizeof() on a Pointer Type

Weakness ID: 467 *(Weakness Variant)* **Status:** Draft

Description

Description Summary

The code calls sizeof() on a malloced pointer type, which always returns the wordsize/8. This can produce an unexpected result if the programmer intended to determine how much memory has been allocated.

Time of Introduction

- Implementation

Applicable Platforms

Languages

C

C++

Common Consequences

Scope	Effect
Integrity	This error can often cause one to allocate a buffer that is much smaller than what is needed, leading to resultant weaknesses such as buffer overflows.

Likelihood of Exploit

High

Demonstrative Examples

Example 1

Care should be taken to ensure sizeof returns the size of the data structure itself, and not the size of the pointer to the data structure.

In this example, sizeof(foo) returns the size of the pointer.

(Bad Code)

Example Languages: **C and C++**

```
double *foo;
...
foo = (double *)malloc(sizeof(foo));
```

In this example, sizeof(*foo) returns the size of the data structure and not the size of the pointer.

(Good Code)

Example Languages: **C and C++**

```
double *foo;
...
foo = (double *)malloc(sizeof(*foo));
```

Example 2

This example defines a fixed username and password. The AuthenticateUser() function is intended to accept a username and a password from an untrusted user, and check to ensure that it matches the username and password. If the username and password match, AuthenticateUser() is intended to indicate that authentication succeeded.

(Bad Code)

```
/* Ignore CWE-259 (hard-coded password) and CWE-309 (use of password system for authentication) for this example. */

char *username = "admin";
char *pass = "password";

int AuthenticateUser(char *inUser, char *inPass) {
```

```
printf("Sizeof username = %d\n", sizeof(username));
printf("Sizeof pass = %d\n", sizeof(pass));

if (strncmp(username, inUser, sizeof(username))) {
printf("Auth failure of username using sizeof\n");
return(AUTH_FAIL);
}
/* Because of CWE-467, the sizeof returns 4 on many platforms and architectures. */
if (! strncmp(pass, inPass, sizeof(pass))) {
printf("Auth success of password using sizeof\n");
return(AUTH_SUCCESS);
}
else {
printf("Auth fail of password using sizeof\n");
return(AUTH_FAIL);
}
}

int main (int argc, char **argv)
{
int authResult;

if (argc < 3) {
ExitError("Usage: Provide a username and password");
}
authResult = AuthenticateUser(argv[1], argv[2]);
if (authResult != AUTH_SUCCESS) {
ExitError("Authentication failed");
}
else {
DoAuthenticatedTask(argv[1]);
}
}
```

In AuthenticateUser(), because sizeof() is applied to a parameter with an array type, the sizeof() call might return 4 on many modern architectures. As a result, the strncmp() call only checks the first four characters of the input password, resulting in a partial comparison (CWE-187), leading to improper authentication (CWE-287).

Because of the partial comparison, any of these passwords would still cause authentication to succeed for the "admin" user:

(Attack)

```
pass5
passABCDEFGH
passWORD
```

Because only 4 characters are checked, this significantly reduces the search space for an attacker, making brute force attacks more feasible.

The same problem also applies to the username, so values such as "adminXYZ" and "administrator" will succeed for the username.

Potential Mitigations

Phase: Implementation

Use expressions such as "sizeof(*pointer)" instead of "sizeof(pointer)", unless you intend to run sizeof() on a pointer type to gain some platform independence or if you are allocating a variable on the stack.

Other Notes

The use of sizeof() on a pointer can sometimes generate useful information. An obvious case is to find out the wordsize on a platform. More often than not, the appearance of sizeof(pointer) indicates a bug.

Weakness Ordinalities

Ordinality	Description
Primary	*(where the weakness exists independent of other weaknesses)*

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	465	Pointer Issues	**Development Concepts (primary)699**
ChildOf	Weakness Class	682	Incorrect Calculation	**Research Concepts (primary)1000**
ChildOf	Category	737	CERT C Secure Coding Section 03 - Expressions (EXP)	**Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734**
ChildOf	Category	740	CERT C Secure Coding Section 06 - Arrays (ARR)	Weaknesses Addressed by the CERT C Secure Coding Standard734
CanPrecede	Weakness Base	131	Incorrect Calculation of Buffer Size	Research Concepts1000

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
CLASP			Use of sizeof() on a pointer type
CERT C Secure Coding	ARR01-C		Do not apply the sizeof operator to a pointer when taking the size of an array
CERT C Secure Coding	EXP01-C		Do not take the size of a pointer to determine the size of the pointed-to type

White Box Definitions

A weakness where code path has:

1. end statement that passes an identity of a dynamically allocated memory resource to a sizeof operator
2. start statement that allocates the dynamically allocated memory resource

References

Robert Seacord. "EXP01-A. Do not take the sizeof a pointer to determine the size of a type". <https://www.securecoding.cert.org/confluence/display/seccode/EXP01-A.+Do+not+take+the+sizeof+a+pointer+to+determine+the+size+of+a+type>.

Content History

Submissions

Submission Date	Submitter	Organization	Source
	CLASP		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-01		KDM Analytics	External
	added/updated white box definitions		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Relationships, Other Notes, Taxonomy Mappings, Weakness Ordinalities		
2008-11-24	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Relationships		

BACK TO TOP

Improper Validation of Array Index

Weakness ID: 129 *(Weakness Base)* **Status:** Draft

Description

Description Summary

The product uses untrusted input when calculating or using an array index, but the product does not validate or incorrectly validates the index to ensure the index references a valid position within the array.

Alternate Terms

out-of-bounds array index

index-out-of-range

array index underflow

Time of Introduction

- Implementation

Applicable Platforms

Languages

C: *(Often)*

C++: *(Often)*

Language-independent

Common Consequences

Scope	Effect
Integrity Availability	Unchecked array indexing will very likely result in the corruption of relevant memory and perhaps instructions, leading to a crash, if the values are outside of the valid memory area.
Integrity	If the memory corrupted is data, rather than instructions, the system will continue to function with improper values.
Confidentiality Integrity	Unchecked array indexing can also trigger out-of-bounds read or write operations, or operations on the wrong objects; i.e., "buffer overflows" are not always the result. This may result in the exposure or modification of sensitive data.
Integrity	If the memory accessible by the attacker can be effectively controlled, it may be possible to execute arbitrary code, as with a standard buffer overflow and possibly without the use of large inputs if a precise index can be controlled.
Integrity Availability Confidentiality	A single fault could allow either an overflow (CWE-788) or underflow (CWE-786) of the array index. What happens next will depend on the type of operation being performed out of bounds, but can expose sensitive information, cause a system crash, or possibly lead to arbitrary code execution.

Likelihood of Exploit

High

Detection Methods

Automated Static Analysis

This weakness can often be detected using automated static analysis tools. Many modern tools use data flow analysis or constraint-based techniques to minimize the number of false positives.

Automated static analysis generally does not account for environmental considerations when reporting out-of-bounds memory operations. This can make it difficult for users to determine which warnings should be investigated first. For example, an analysis tool might report array index errors that originate from command line arguments in a program that is not expected to run with setuid or other special privileges.

Effectiveness: High

This is not a perfect solution, since 100% accuracy and coverage are not feasible.

Automated Dynamic Analysis

This weakness can be detected using dynamic tools and techniques that interact with the software using large test suites with many diverse inputs, such as fuzz testing (fuzzing), robustness testing, and fault injection. The software's operation may slow down, but it should not become unstable, crash, or generate incorrect results.

Black Box

Black box methods might not get the needed code coverage within limited time constraints, and a dynamic test might not produce any noticeable side effects even if it is successful.

Demonstrative Examples

Example 1

The following C/C++ example retrieves the sizes of messages for a pop3 mail server. The message sizes are retrieved from a socket that returns in a buffer the message number and the message size, the message number (num) and size (size) are extracted from the buffer and the message size is placed into an array using the message number for the array index.

(Bad Code)

Example Language: **C**

```
/* capture the sizes of all messages */
int getsizes(int sock, int count, int *sizes) {
...
char buf[BUFFER_SIZE];
int ok;
int num, size;

// read values from socket and added to sizes array
while ((ok = gen_recv(sock, buf, sizeof(buf))) == 0)
{

// continue read from socket until buf only contains '.'
if (DOTLINE(buf))
break;
else if (sscanf(buf, "%d %d", &num, &size) == 2)
sizes[num - 1] = size;
}
...
}
```

In this example the message number retrieved from the buffer could be a value that is outside the allowable range of indices for the array and could possibly be a negative number. Without proper validation of the value to be used for the array index an array overflow could occur and could potentially lead to unauthorized access to memory addresses and system crashes. The value of the array index should be validated to ensure that it is within the allowable range of indices for the array as in the following code.

(Good Code)

Example Language: **C**

```
/* capture the sizes of all messages */
int getsizes(int sock, int count, int *sizes) {
...
char buf[BUFFER_SIZE];
int ok;
int num, size;

// read values from socket and added to sizes array
while ((ok = gen_recv(sock, buf, sizeof(buf))) == 0)
{

// continue read from socket until buf only contains '.'
if (DOTLINE(buf))
```

```
break;
else if (sscanf(buf, "%d %d", &num, &size) == 2) {
if (num > 0 && num <= (unsigned)count)
sizes[num - 1] = size;
else
/* warn about possible attempt to induce buffer overflow */
report(stderr, "Warning: ignoring bogus data for message sizes returned by server.\n");
}
}
...
}
```

Example 2

In the code snippet below, an unchecked integer value is used to reference an object in an array.

(Bad Code)

Example Language: **Java**

```
public String getValue(int index) {
return array[index];
}
```

If index is outside of the range of the array, this may result in an ArrayIndexOutOfBounds Exception being raised.

Example 3

In the following Java example the method displayProductSummary is called from a Web service servlet to retrieve product summary information for display to the user. The servlet obtains the integer value of the product number from the user and passes it to the displayProductSummary method. The displayProductSummary method passes the integer value of the product number to the getProductSummary method which obtains the product summary from the array object containing the project summaries using the integer value of the product number as the array index.

(Bad Code)

Example Language: **Java**

```
// Method called from servlet to obtain product information
public String displayProductSummary(int index) {

String productSummary = new String("");

try {
String productSummary = getProductSummary(index);

} catch (Exception ex) {...}

return productSummary;
}

public String getProductSummary(int index) {
return products[index];
}
```

In this example the integer value used as the array index that is provided by the user may be outside the allowable range of indices for the array which may provide unexpected results or may comes the application to fail. The integer value used for the array index should be validated to ensure that it is within the allowable range of indices for the array as in the following code.

(Good Code)

Example Language: **Java**

```
// Method called from servlet to obtain product information
public String displayProductSummary(int index) {

String productSummary = new String("");
```

```
try {
String productSummary = getProductSummary(index);

} catch (Exception ex) {...}

return productSummary;
}

public String getProductSummary(int index) {
String productSummary = "";

if ((index >= 0) && (index < MAX_PRODUCTS)) {
productSummary = products[index];
}
else {
System.err.println("index is out of bounds");
throw new IndexOutOfBoundsException();
}

return productSummary;
}
```

An alternative in Java would be to use one of the collection objects such as ArrayList that will automatically generate an exception if an attempt is made to access an array index that is out of bounds.

(Good Code)

Example Language: **Java**

```
ArrayList productArray = new ArrayList(MAX_PRODUCTS);
...
try {
productSummary = (String) productArray.get(index);
} catch (IndexOutOfBoundsException ex) {...}
```

Observed Examples

Reference	Description
CVE-2005-0369	large ID in packet used as array index
CVE-2001-1009	negative array index as argument to POP LIST command
CVE-2003-0721	Integer signedness error leads to negative array index
CVE-2004-1189	product does not properly track a count and a maximum number, which can lead to resultant array index overflow.
CVE-2007-5756	chain: device driver for packet-capturing software allows access to an unintended IOCTL with resultant array index error.

Potential Mitigations

Phase: Architecture and Design

Strategies: Input Validation; Libraries or Frameworks

Use an input validation framework such as Struts or the OWASP ESAPI Validation API. If you use Struts, be mindful of weaknesses covered by the CWE-101 category.

Phase: Architecture and Design

For any security checks that are performed on the client side, ensure that these checks are duplicated on the server side, in order to avoid CWE-602. Attackers can bypass the client-side checks by modifying values after the checks have been performed, or by changing the client to remove the client-side checks entirely. Then, these modified values would be submitted to the server.

Even though client-side checks provide minimal benefits with respect to server-side security, they are still useful. First, they can support intrusion detection. If the server receives input that should have been rejected by the client, then it may be an indication of an attack. Second, client-side error-checking can provide helpful feedback to the user about the expectations for valid input. Third, there may be a reduction in server-side processing time for accidental input errors, although this is typically a small savings.

Phase: Requirements

Strategy: Language Selection

Use a language with features that can automatically mitigate or eliminate out-of-bounds indexing errors.

For example, Ada allows the programmer to constrain the values of a variable and languages such as Java and Ruby will allow the programmer to handle exceptions when an out-of-bounds index is accessed.

Phase: Implementation

Strategy: Input Validation

Assume all input is malicious. Use an "accept known good" input validation strategy (i.e., use a whitelist). Reject any input that does not strictly conform to specifications, or transform it into something that does. Use a blacklist to reject any unexpected inputs and detect potential attacks.

When accessing a user-controlled array index, use a stringent range of values that are within the target array. Make sure that you do not allow negative values to be used. That is, verify the minimum as well as the maximum of the range of acceptable values.

Phase: Implementation

Be especially careful to validate your input when you invoke code that crosses language boundaries, such as from an interpreted language to native code. This could create an unexpected interaction between the language boundaries. Ensure that you are not violating any of the expectations of the language with which you are interfacing. For example, even though Java may not be susceptible to buffer overflows, providing a large argument in a call to native code might trigger an overflow.

Weakness Ordinalities

Ordinality	Description
Resultant	The most common condition situation leading to unchecked array indexing is the use of loop index variables as buffer indexes. If the end condition for the loop is subject to a flaw, the index can grow or shrink unbounded, therefore causing a buffer overflow or underflow. Another common situation leading to this condition is the use of a function's return value, or the resulting value of a calculation directly as an index in to a buffer.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	20	Improper Input Validation	**Development Concepts (primary)699 Research Concepts (primary)1000**
ChildOf	Category	189	Numeric Errors	Development Concepts699
ChildOf	Category	633	Weaknesses that Affect Memory	**Resource-specific Weaknesses (primary)631**
ChildOf	Category	738	CERT C Secure Coding Section 04 - Integers (INT)	**Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734**
ChildOf	Category	740	CERT C Secure Coding Section 06 - Arrays (ARR)	Weaknesses Addressed by the CERT C Secure Coding Standard734
ChildOf	Category	802	2010 Top 25 - Risky Resource Management	**Weaknesses in the 2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800**
CanPrecede	Weakness Class	119	Failure to Constrain Operations within the Bounds of a Memory Buffer	Research Concepts1000
CanPrecede	Weakness Variant	789	Uncontrolled Memory Allocation	Research Concepts1000
PeerOf	Weakness Base	124	Buffer Underwrite ('Buffer Underflow')	Research Concepts1000

Theoretical Notes

An improperly validated array index might lead directly to the always-incorrect behavior of "access of array using out-of-bounds index."

Affected Resources

- Memory

f Causal Nature

Explicit

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
CLASP			Unchecked array indexing
PLOVER			INDEX - Array index overflow
CERT C Secure Coding	ARR00-C		Understand how arrays work
CERT C Secure Coding	ARR30-C		Guarantee that array indices are within the valid range
CERT C Secure Coding	ARR38-C		Do not add or subtract an integer to a pointer if the resulting value does not refer to a valid array element
CERT C Secure Coding	INT32-C		Ensure that operations on signed integers do not result in overflow

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
100	Overflow Buffers	

References

[REF-11] M. Howard and D. LeBlanc. "Writing Secure Code". Chapter 5, "Array Indexing Errors" Page 144. 2nd Edition. Microsoft. 2002.

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	CLASP		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Sean Eidemiller	Cigital	External
	added/updated demonstrative examples		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Alternate Terms, Applicable Platforms, Common Consequences, Relationships, Other Notes, Taxonomy Mappings, Weakness Ordinalities		
2008-11-24	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2009-01-12	CWE Content Team	MITRE	Internal
	updated Common Consequences		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Description, Name, Relationships		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Observed Examples, Other Notes, Potential Mitigations, Theoretical Notes, Weakness Ordinalities		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Demonstrative Examples, Detection Factors, Likelihood of Exploit, Potential Mitigations, References, Related Attack Patterns, Relationships		
2010-04-05	CWE Content Team	MITRE	Internal
	updated Related Attack Patterns		
Previous Entry Names			
Change Date	**Previous Entry Name**		
2009-10-29	Unchecked Array Indexing		

BACK TO TOP

Improper Access Control (Authorization)

Weakness ID: 285 *(Weakness Class)* **Status:** Draft

Description

Description Summary

The software does not perform or incorrectly performs access control checks across all potential execution paths.

Extended Description

When access control checks are not applied consistently - or not at all - users are able to access data or perform actions that they should not be allowed to perform. This can lead to a wide range of problems, including information leaks, denial of service, and arbitrary code execution.

Alternate Terms

AuthZ:	"AuthZ" is typically used as an abbreviation of "authorization" within the web application security community. It is also distinct from "AuthC," which is an abbreviation of "authentication." The use of "Auth" as an abbreviation is discouraged, since it could be used for either authentication or authorization.

Time of Introduction

- Architecture and Design
- Implementation
- Operation

Applicable Platforms

Languages

Language-independent

Technology Classes

Web-Server: *(Often)*

Database-Server: *(Often)*

Modes of Introduction

A developer may introduce authorization weaknesses because of a lack of understanding about the underlying technologies. For example, a developer may assume that attackers cannot modify certain inputs such as headers or cookies.

Authorization weaknesses may arise when a single-user application is ported to a multi-user environment.

Common Consequences

Scope	Effect
Confidentiality	An attacker could read sensitive data, either by reading the data directly from a data store that is not properly restricted, or by accessing insufficiently-protected, privileged functionality to read the data.
Integrity	An attacker could modify sensitive data, either by writing the data directly to a data store that is not properly restricted, or by accessing insufficiently-protected, privileged functionality to write the data.
Integrity	An attacker could gain privileges by modifying or reading critical data directly, or by accessing insufficiently-protected, privileged functionality.

Likelihood of Exploit

High

Detection Methods

Automated Static Analysis

Automated static analysis is useful for detecting commonly-used idioms for authorization. A tool may be able to analyze related configuration files, such as .htaccess in Apache web servers, or detect the usage of commonly-used authorization libraries.

Generally, automated static analysis tools have difficulty detecting custom authorization schemes. In addition, the software's design may include some functionality that is accessible to any user and does not require an authorization check; an automated technique that detects the absence of authorization may report false positives.

Effectiveness: Limited

Automated Dynamic Analysis

Automated dynamic analysis may find many or all possible interfaces that do not require authorization, but manual analysis is required to determine if the lack of authorization violates business logic

Manual Analysis

This weakness can be detected using tools and techniques that require manual (human) analysis, such as penetration testing, threat modeling, and interactive tools that allow the tester to record and modify an active session.

Specifically, manual static analysis is useful for evaluating the correctness of custom authorization mechanisms.

Effectiveness: Moderate

These may be more effective than strictly automated techniques. This is especially the case with weaknesses that are related to design and business rules. However, manual efforts might not achieve desired code coverage within limited time constraints.

Demonstrative Examples

Example 1

The following program could be part of a bulletin board system that allows users to send private messages to each other. This program intends to authenticate the user before deciding whether a private message should be displayed. Assume that LookupMessageObject() ensures that the $id argument is numeric, constructs a filename based on that id, and reads the message details from that file. Also assume that the program stores all private messages for all users in the same directory.

(Bad Code)

Example Language: **Perl**

```
sub DisplayPrivateMessage {
my($id) = @_;
my $Message = LookupMessageObject($id);
print "From: " . encodeHTML($Message->{from}) . "<br>\n";
print "Subject: " . encodeHTML($Message->{subject}) . "\n";
print "<hr>\n";
print "Body: " . encodeHTML($Message->{body}) . "\n";
}

my $q = new CGI;
# For purposes of this example, assume that CWE-309 and
# CWE-523 do not apply.
if (! AuthenticateUser($q->param('username'), $q->param('password'))) {
ExitError("invalid username or password");
}

my $id = $q->param('id');
DisplayPrivateMessage($id);
```

While the program properly exits if authentication fails, it does not ensure that the message is addressed to the user. As a result, an authenticated attacker could provide any arbitrary identifier and read private messages that were intended for other users.

One way to avoid this problem would be to ensure that the "to" field in the message object matches the username of the authenticated user.

Observed Examples

Reference	Description
CVE-2009-3168	Web application does not restrict access to admin scripts, allowing authenticated users to reset administrative passwords.

CVE-2009-2960	Web application does not restrict access to admin scripts, allowing authenticated users to modify passwords of other users.
CVE-2009-3597	Web application stores database file under the web root with insufficient access control (CWE-219), allowing direct request.
CVE-2009-2282	Terminal server does not check authorization for guest access.
CVE-2009-3230	Database server does not use appropriate privileges for certain sensitive operations.
CVE-2009-2213	Gateway uses default "Allow" configuration for its authorization settings.
CVE-2009-0034	Chain: product does not properly interpret a configuration option for a system group, allowing users to gain privileges.
CVE-2008-6123	Chain: SNMP product does not properly parse a configuration option for which hosts are allowed to connect, allowing unauthorized IP addresses to connect.
CVE-2008-5027	System monitoring software allows users to bypass authorization by creating custom forms.
CVE-2008-7109	Chain: reliance on client-side security (CWE-602) allows attackers to bypass authorization using a custom client.
CVE-2008-3424	Chain: product does not properly handle wildcards in an authorization policy list, allowing unintended access.
CVE-2009-3781	Content management system does not check access permissions for private files, allowing others to view those files.
CVE-2008-4577	ACL-based protection mechanism treats negative access rights as if they are positive, allowing bypass of intended restrictions.
CVE-2008-6548	Product does not check the ACL of a page accessed using an "include" directive, allowing attackers to read unauthorized files.
CVE-2007-2925	Default ACL list for a DNS server does not set certain ACLs, allowing unauthorized DNS queries.
CVE-2006-6679	Product relies on the X-Forwarded-For HTTP header for authorization, allowing unintended access by spoofing the header.
CVE-2005-3623	OS kernel does not check for a certain privilege before setting ACLs for files.
CVE-2005-2801	Chain: file-system code performs an incorrect comparison (CWE-697), preventing defauls ACLs from being properly applied.
CVE-2001-1155	Chain: product does not properly check the result of a reverse DNS lookup because of operator precedence (CWE-783), allowing bypass of DNS-based access restrictions.

Potential Mitigations

Phase: Architecture and Design

Divide your application into anonymous, normal, privileged, and administrative areas. Reduce the attack surface by carefully mapping roles with data and functionality. Use role-based access control (RBAC) to enforce the roles at the appropriate boundaries.

Note that this approach may not protect against horizontal authorization, i.e., it will not protect a user from attacking others with the same role.

Phase: Architecture and Design

Ensure that you perform access control checks related to your business logic. These checks may be different than the access control checks that you apply to more generic resources such as files, connections, processes, memory, and database records. For example, a database may restrict access for medical records to a specific database user, but each record might only be intended to be accessible to the patient and the patient's doctor.

Phase: Architecture and Design

Strategy: Libraries or Frameworks

Use a vetted library or framework that does not allow this weakness to occur or provides constructs that make this weakness

easier to avoid.

For example, consider using authorization frameworks such as the JAAS Authorization Framework and the OWASP ESAPI Access Control feature.

Phase: Architecture and Design

For web applications, make sure that the access control mechanism is enforced correctly at the server side on every page. Users should not be able to access any unauthorized functionality or information by simply requesting direct access to that page.

One way to do this is to ensure that all pages containing sensitive information are not cached, and that all such pages restrict access to requests that are accompanied by an active and authenticated session token associated with a user who has the required permissions to access that page.

Phases: System Configuration; Installation

Use the access control capabilities of your operating system and server environment and define your access control lists accordingly. Use a "default deny" policy when defining these ACLs.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	254	Security Features	**Seven Pernicious Kingdoms (primary)700**
ChildOf	Weakness Class	284	Access Control (Authorization) Issues	**Development Concepts (primary)699 Research Concepts (primary)1000**
ChildOf	Category	721	OWASP Top Ten 2007 Category A10 - Failure to Restrict URL Access	**Weaknesses in OWASP Top Ten (2007) (primary)629**
ChildOf	Category	723	OWASP Top Ten 2004 Category A2 - Broken Access Control	**Weaknesses in OWASP Top Ten (2004) (primary)711**
ChildOf	Category	753	2009 Top 25 - Porous Defenses	**Weaknesses in the 2009 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)750**
ChildOf	Category	803	2010 Top 25 - Porous Defenses	**Weaknesses in the 2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800**
ParentOf	Weakness Variant	219	Sensitive Data Under Web Root	**Research Concepts (primary)1000**
ParentOf	Weakness Base	551	Incorrect Behavior Order: Authorization Before Parsing and Canonicalization	**Development Concepts (primary)699** Research Concepts1000
ParentOf	Weakness Class	638	Failure to Use Complete Mediation	Research Concepts1000
ParentOf	Weakness Base	804	Guessable CAPTCHA	**Development Concepts (primary)699 Research Concepts (primary)1000**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
7 Pernicious Kingdoms			Missing Access Control
OWASP Top Ten 2007	A10	CWE More Specific	Failure to Restrict URL Access
OWASP Top Ten 2004	A2	CWE More Specific	Broken Access Control

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
1	Accessing Functionality Not Properly Constrained by ACLs	
13	Subverting Environment Variable Values	

17	Accessing, Modifying or Executing Executable Files
87	Forceful Browsing
39	Manipulating Opaque Client-based Data Tokens
45	Buffer Overflow via Symbolic Links
51	Poison Web Service Registry
59	Session Credential Falsification through Prediction
60	Reusing Session IDs (aka Session Replay)
77	Manipulating User-Controlled Variables
76	Manipulating Input to File System Calls
104	Cross Zone Scripting

References

NIST. "Role Based Access Control and Role Based Security". <http://csrc.nist.gov/groups/SNS/rbac/>.

[REF-11] M. Howard and D. LeBlanc. "Writing Secure Code". Chapter 4, "Authorization" Page 114; Chapter 6, "Determining Appropriate Access Control" Page 171. 2nd Edition. Microsoft. 2002.

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	7 Pernicious Kingdoms		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-15		Veracode	External
	Suggested OWASP Top Ten 2004 mapping		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Relationships, Other Notes, Taxonomy Mappings		
2009-01-12	CWE Content Team	MITRE	Internal
	updated Common Consequences, Description, Likelihood of Exploit, Name, Other Notes, Potential Mitigations, References, Relationships		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Potential Mitigations		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Description, Related Attack Patterns		
2009-07-27	CWE Content Team	MITRE	Internal
	updated Relationships		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Type		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Demonstrative Examples, Detection Factors, Modes of Introduction, Observed Examples, Relationships		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Alternate Terms, Detection Factors, Potential Mitigations, References, Relationships		
2010-04-05	CWE Content Team	MITRE	Internal
	updated Potential Mitigations		
Previous Entry Names			
Change Date	**Previous Entry Name**		
2009-01-12	Missing or Inconsistent Access Control		

BACK TO TOP

Incorrect Permission Assignment for Critical Resource

Weakness ID: 732 *(Weakness Class)* **Status:** Draft

Description

Description Summary

The software specifies permissions for a security-critical resource in a way that allows that resource to be read or modified by unintended actors.

Extended Description

When a resource is given a permissions setting that provides access to a wider range of actors than required, it could lead to the disclosure of sensitive information, or the modification of that resource by unintended parties. This is especially dangerous when the resource is related to program configuration, execution or sensitive user data.

Time of Introduction

- Architecture and Design
- Implementation
- Installation
- Operation

Applicable Platforms

Languages

Language-independent

Modes of Introduction

The developer may set loose permissions in order to minimize problems when the user first runs the program, then create documentation stating that permissions should be tightened. Since system administrators and users do not always read the documentation, this can result in insecure permissions being left unchanged.

The developer might make certain assumptions about the environment in which the software runs - e.g., that the software is running on a single-user system, or the software is only accessible to trusted administrators. When the software is running in a different environment, the permissions become a problem.

Common Consequences

Scope	Effect
Confidentiality	An attacker may be able to read sensitive information from the associated resource, such as credentials or configuration information stored in a file.
Integrity	An attacker may be able to modify critical properties of the associated resource to gain privileges, such as replacing a world-writable executable with a Trojan horse.
Availability	An attacker may be able to destroy or corrupt critical data in the associated resource, such as deletion of records from a database.

Likelihood of Exploit

Medium to High

Detection Methods

Automated Static Analysis

Automated static analysis may be effective in detecting permission problems for system resources such as files, directories, shared memory, device interfaces, etc. Automated techniques may be able to detect the use of library functions that modify permissions, then analyze function calls for arguments that contain potentially insecure values.

However, since the software's intended security policy might allow loose permissions for certain operations (such as publishing a file on a web server), automated static analysis may produce some false positives - i.e., warnings that do not have any security consequences or require any code changes.

When custom permissions models are used - such as defining who can read messages in a particular forum in a bulletin board system - these can be difficult to detect using automated static analysis. It may be possible to define custom signatures that

identify any custom functions that implement the permission checks and assignments.

Automated Dynamic Analysis

Automated dynamic analysis may be effective in detecting permission problems for system resources such as files, directories, shared memory, device interfaces, etc.

However, since the software's intended security policy might allow loose permissions for certain operations (such as publishing a file on a web server), automated dynamic analysis may produce some false positives - i.e., warnings that do not have any security consequences or require any code changes.

When custom permissions models are used - such as defining who can read messages in a particular forum in a bulletin board system - these can be difficult to detect using automated dynamic analysis. It may be possible to define custom signatures that identify any custom functions that implement the permission checks and assignments.

Manual Static Analysis

Manual static analysis may be effective in detecting the use of custom permissions models and functions. The code could then be examined to identifying usage of the related functions. Then the human analyst could evaluate permission assignments in the context of the intended security model of the software.

Manual Dynamic Analysis

Manual dynamic analysis may be effective in detecting the use of custom permissions models and functions. The program could then be executed with a focus on exercising code paths that are related to the custom permissions. Then the human analyst could evaluate permission assignments in the context of the intended security model of the software.

Fuzzing

Fuzzing is not effective in detecting this weakness.

Demonstrative Examples

Example 1

The following code sets the umask of the process to 0 before creating a file and writing "Hello world" into the file.

(Bad Code)

Example Language: **C**

```
#define OUTFILE "hello.out"

umask(0);
FILE *out;
/* Ignore CWE-59 (link following) for brevity */
out = fopen(OUTFILE, "w");
if (out) {
fprintf(out, "hello world!\n");
fclose(out);
}
```

After running this program on a UNIX system, running the "ls -l" command might return the following output:

(Result)

```
-rw-rw-rw- 1 username 13 Nov 24 17:58 hello.out
```

The "rw-rw-rw-" string indicates that the owner, group, and world (all users) can read the file and write to it.

Example 2

The following code snippet might be used as a monitor to periodically record whether a web site is alive. To ensure that the file can always be modified, the code uses chmod() to make the file world-writable.

(Bad Code)

Example Language: **Perl**

```
$fileName = "secretFile.out";

if (-e $fileName) {
chmod 0777, $fileName;
}
```

```
my $outFH;
if (! open($outFH, ">>$fileName")) {
ExitError("Couldn't append to $fileName: $!");
}
my $dateString = FormatCurrentTime();
my $status = IsHostAlive("cwe.mitre.org");
print $outFH "$dateString cwe status: $status!\n";
close($outFH);
```

The first time the program runs, it might create a new file that inherits the permissions from its environment. A file listing might look like:

(Result)

```
-rw-r--r-- 1 username 13 Nov 24 17:58 secretFile.out
```

This listing might occur when the user has a default umask of 022, which is a common setting. Depending on the nature of the file, the user might not have intended to make it readable by everyone on the system.

The next time the program runs, however - and all subsequent executions - the chmod will set the file's permissions so that the owner, group, and world (all users) can read the file and write to it:

(Result)

```
-rw-rw-rw- 1 username 13 Nov 24 17:58 secretFile.out
```

Perhaps the programmer tried to do this because a different process uses different permissions that might prevent the file from being updated.

Example 3

The following command recursively sets world-readable permissions for a directory and all of its children:

(Bad Code)

Example Language: **Shell**

```
chmod -R ugo+r DIRNAME
```

If this command is run from a program, the person calling the program might not expect that all the files under the directory will be world-readable. If the directory is expected to contain private data, this could become a security problem.

Observed Examples

Reference	Description
CVE-2009-3482	Anti-virus product sets insecure "Everyone: Full Control" permissions for files under the "Program Files" folder, allowing attackers to replace executables with Trojan horses.
CVE-2009-3897	Product creates directories with 0777 permissions at installation, allowing users to gain privileges and access a socket used for authentication.
CVE-2009-3489	Photo editor installs a service with an insecure security descriptor, allowing users to stop or start the service, or execute commands as SYSTEM.
CVE-2009-3289	Library function copies a file to a new target and uses the source file's permissions for the target, which is incorrect when the source file is a symbolic link, which typically has 0777 permissions.
CVE-2009-0115	Device driver uses world-writable permissions for a socket file, allowing attackers to inject arbitrary commands.
CVE-2009-1073	LDAP server stores a cleartext password in a world-readable file.
CVE-2009-0141	Terminal emulator creates TTY devices with world-writable permissions, allowing an attacker to write to the terminals of other users.

CVE-2008-0662	VPN product stores user credentials in a registry key with "Everyone: Full Control" permissions, allowing attackers to steal the credentials.
CVE-2008-0322	Driver installs its device interface with "Everyone: Write" permissions.
CVE-2009-3939	Driver installs a file with world-writable permissions.
CVE-2009-3611	Product changes permissions to 0777 before deleting a backup; the permissions stay insecure for subsequent backups.
CVE-2007-6033	Product creates a share with "Everyone: Full Control" permissions, allowing arbitrary program execution.
CVE-2007-5544	Product uses "Everyone: Full Control" permissions for memory-mapped files (shared memory) in inter-process communication, allowing attackers to tamper with a session.
CVE-2005-4868	Database product uses read/write permissions for everyone for its shared memory, allowing theft of credentials.
CVE-2004-1714	Security product uses "Everyone: Full Control" permissions for its configuration files.
CVE-2001-0006	"Everyone: Full Control" permissions assigned to a mutex allows users to disable network connectivity.
CVE-2002-0969	Chain: database product contains buffer overflow that is only reachable through a .ini configuration file - which has "Everyone: Full Control" permissions.

Potential Mitigations

Phase: Implementation

When using a critical resource such as a configuration file, check to see if the resource has insecure permissions (such as being modifiable by any regular user), and generate an error or even exit the software if there is a possibility that the resource could have been modified by an unauthorized party.

Phase: Architecture and Design

Divide your application into anonymous, normal, privileged, and administrative areas. Reduce the attack surface by carefully defining distinct user groups, privileges, and/or roles. Map these against data, functionality, and the related resources. Then set the permissions accordingly. This will allow you to maintain more fine-grained control over your resources.

Phases: Implementation; Installation

During program startup, explicitly set the default permissions or umask to the most restrictive setting possible. Also set the appropriate permissions during program installation. This will prevent you from inheriting insecure permissions from any user who installs or runs the program.

Phase: System Configuration

For all configuration files, executables, and libraries, make sure that they are only readable and writable by the software's administrator.

Phase: Documentation

Do not suggest insecure configuration changes in your documentation, especially if those configurations can extend to resources and other software that are outside the scope of your own software.

Phase: Installation

Do not assume that the system administrator will manually change the configuration to the settings that you recommend in the manual.

Phase: Testing

Use tools and techniques that require manual (human) analysis, such as penetration testing, threat modeling, and interactive tools that allow the tester to record and modify an active session. These may be more effective than strictly automated techniques. This is especially the case with weaknesses that are related to design and business rules.

Phase: Testing

Use monitoring tools that examine the software's process as it interacts with the operating system and the network. This technique is useful in cases when source code is unavailable, if the software was not developed by you, or if you want to verify that the build phase did not introduce any new weaknesses. Examples include debuggers that directly attach to the running process; system-call tracing utilities such as truss (Solaris) and strace (Linux); system activity monitors such as FileMon, RegMon, Process Monitor, and other Sysinternals utilities (Windows); and sniffers and protocol analyzers that monitor network traffic.

Attach the monitor to the process and watch for library functions or system calls on OS resources such as files, directories, and shared memory. Examine the arguments to these calls to infer which permissions are being used.

Note that this technique is only useful for permissions issues related to system resources. It is not likely to detect application-level business rules that are related to permissions, such as if a user of a blog system marks a post as "private," but the blog system inadvertently marks it as "public."

Phases: Testing; System Configuration

Ensure that your software runs properly under the Federal Desktop Core Configuration (FDCC) or an equivalent hardening configuration guide, which many organizations use to limit the attack surface and potential risk of deployed software.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	275	Permission Issues	**Development Concepts (primary)699**
ChildOf	Weakness Class	668	Exposure of Resource to Wrong Sphere	**Research Concepts (primary)1000**
ChildOf	Category	753	2009 Top 25 - Porous Defenses	**Weaknesses in the 2009 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)750**
ChildOf	Category	803	2010 Top 25 - Porous Defenses	**Weaknesses in the 2010 CWE/SANS Top 25 Most Dangerous Programming Errors (primary)800**
RequiredBy	Compound Element: Composite	689	Permission Race Condition During Resource Copy	Research Concepts1000
ParentOf	Weakness Variant	276	Incorrect Default Permissions	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	277	Insecure Inherited Permissions	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	278	Insecure Preserved Inherited Permissions	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	279	Incorrect Execution-Assigned Permissions	**Research Concepts (primary)1000**
ParentOf	Weakness Base	281	Improper Preservation of Permissions	**Research Concepts (primary)1000**

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	*(CAPEC Version: 1.5)*
232	Exploitation of Privilege/Trust	
1	Accessing Functionality Not Properly Constrained by ACLs	
17	Accessing, Modifying or Executing Executable Files	
60	Reusing Session IDs (aka Session Replay)	
61	Session Fixation	
62	Cross Site Request Forgery (aka Session Riding)	
122	Exploitation of Authorization	
180	Exploiting Incorrectly Configured Access Control Security Levels	
234	Hijacking a privileged process	

References

Mark Dowd, John McDonald and Justin Schuh. "The Art of Software Security Assessment". Chapter 9, "File Permissions." Page 495.. 1st Edition. Addison Wesley. 2006.

John Viega and Gary McGraw. "Building Secure Software". Chapter 8, "Access Control." Page 194.. 1st Edition. Addison-Wesley. 2002.

Maintenance Notes

The relationships between privileges, permissions, and actors (e.g. users and groups) need further refinement within the Research view. One complication is that these concepts apply to two different pillars, related to control of resources (CWE-664) and protection mechanism failures (CWE-396).

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
2008-09-08			Internal CWE Team
	new weakness-focused entry for Research view.		
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2009-01-12	CWE Content Team	MITRE	Internal
	updated Description, Likelihood of Exploit, Name, Potential Mitigations, Relationships		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Potential Mitigations, Related Attack Patterns		
2009-05-27	CWE Content Team	MITRE	Internal
	updated Name		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Demonstrative Examples, Detection Factors, Modes of Introduction, Observed Examples, Potential Mitigations, References		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Relationships		
2010-04-05	CWE Content Team	MITRE	Internal
	updated Potential Mitigations, Related Attack Patterns		
Previous Entry Names			
Change Date	**Previous Entry Name**		
2009-01-12	Insecure Permission Assignment for Resource		
2009-05-27	Insecure Permission Assignment for Critical Resource		

BACK TO TOP

Exposure of System Data to Unauthorized Control Sphere

Risk

What might happen

System data can provide attackers with valuable insights on systems and services they are targeting - any type of system data, from service version to operating system fingerprints, can assist attackers to hone their attack, correlate data with known vulnerabilities or focus efforts on developing new attacks against specific technologies.

Cause

How does it happen

System data is read and subsequently exposed where it might be read by untrusted entities.

General Recommendations

How to avoid it

Consider the implications of exposure of the specified input, and expected level of access to the specified output. If not required, consider removing this code, or modifying exposed information to exclude potentially sensitive system data.

Source Code Examples

Java

Leaking Environment Variables in JSP Web-Page

```
String envVarValue = System.getenv(envVar);
if (envVarValue == null) {
    out.println("Environment variable is not defined:");
    out.println(System.getenv());
} else {
    //[..]
};
```

TOCTOU

Risk

What might happen

At best, a Race Condition may cause errors in accuracy, overidden values or unexpected behavior that may result in denial-of-service. At worst, it may allow attackers to retrieve data or bypass security processes by replaying a controllable Race Condition until it plays out in their favor.

Cause

How does it happen

Race Conditions occur when a public, single instance of a resource is used by multiple concurrent logical processes. If the these logical processes attempt to retrieve and update the resource without a timely management system, such as a lock, a Race Condition will occur.

An example for when a Race Condition occurs is a resource that may return a certain value to a process for further editing, and then updated by a second process, resulting in the original process' data no longer being valid. Once the original process edits and updates the incorrect value back into the resource, the second process' update has been overwritten and lost.

General Recommendations

How to avoid it

When sharing resources between concurrent processes across the application ensure that these resources are either thread-safe, or implement a locking mechanism to ensure expected concurrent activity.

Source Code Examples

Java

Different Threads Increment and Decrement The Same Counter Repeatedly, Resulting in a Race Condition

```
	public static int counter = 0;
	public static void start() throws InterruptedException {
		incrementCounter ic;
		decrementCounter dc;
		while(counter == 0) {
			counter = 0;
			ic = new incrementCounter();
			dc = new decrementCounter();
			ic.start();
			dc.start();
			ic.join();
			dc.join();
		}
		System.out.println(counter); //Will stop and return either -1 or 1 due to race
condition over counter
	}

	public static class incrementCounter extends Thread {
	    public void run() {
		counter++;
	    }
```

```
    }

    public static class decrementCounter extends Thread {
        public void run() {
           counter--;
        }
    }
```

Different Threads Increment and Decrement The Same Thread-Safe Counter Repeatedly, Never Resulting in a Race Condition

```
    public static int counter = 0;
    public static Object lock = new Object();

    public static void start() throws InterruptedException {
           incrementCounter ic;
           decrementCounter dc;
           while(counter == 0) { // because of proper locking, this condition is never false
                  counter = 0;
                  ic = new incrementCounter();
                  dc = new decrementCounter();
                  ic.start();
                  dc.start();
                  ic.join();
                  dc.join();
           }
           System.out.println(counter); // Never reached
    }

    public static class incrementCounter extends Thread {
        public void run() {
           synchronized (lock) {
                  counter++;
           }
        }
    }

    public static class decrementCounter extends Thread {
        public void run() {
           synchronized (lock) {
                  counter--;
           }
        }
    }
```

Information Leak Through Comments

Weakness ID: 615 *(Weakness Variant)* **Status:** Incomplete

Description

Description Summary

While adding general comments is very useful, some programmers tend to leave important data, such as: filenames related to the web application, old links or links which were not meant to be browsed by users, old code fragments, etc.

Extended Description

An attacker who finds these comments can map the application's structure and files, expose hidden parts of the site, and study the fragments of code to reverse engineer the application, which may help develop further attacks against the site.

Time of Introduction

- Implementation

Demonstrative Examples

Example 1

The following comment, embedded in a JSP, will be displayed in the resulting HTML output.

(Bad Code)

Example Languages: **HTML and JSP**

```
<!-- FIXME: calling this with more than 30 args kills the JDBC server -->
```

Observed Examples

Reference	Description
CVE-2007-6197	Version numbers and internal hostnames leaked in HTML comments.
CVE-2007-4072	CMS places full pathname of server in HTML comment.
CVE-2009-2431	blog software leaks real username in HTML comment.

Potential Mitigations

Remove comments which have sensitive information about the design/implementation of the application. Some of the comments may be exposed to the user and affect the security posture of the application.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Variant	540	Information Leak Through Source Code	**Development Concepts (primary)699 Research Concepts (primary)1000**

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	Anonymous Tool Vendor (under NDA)		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Sean Eidemiller	Cigital	External
	added/updated demonstrative examples		
2008-07-01	Eric Dalci	Cigital	External
	updated Potential Mitigations, Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2008-10-14	CWE Content Team	MITRE	Internal
	updated Description		
2009-03-10	CWE Content Team	MITRE	Internal

	updated Demonstrative Examples		
2009-07-27	CWE Content Team	MITRE	Internal
	updated Observed Examples, Taxonomy Mappings		

BACK TO TOP

Scanned Languages

Language	Hash Number	Change Date
CPP	4541647240435660	6/19/2024
Common	0105849645654507	6/19/2024