PolarDB-for-PostgreSQL-1 Scan Report

Project Name	PolarDB-for-PostgreSQL-1
Scan Start	Friday, June 21, 2024 1:07:35 PM
Preset	Checkmarx Default
Scan Time	01h:27m:23s
Lines Of Code Scanned	170556
Files Scanned	88
Report Creation Time	Friday, June 21, 2024 3:22:04 PM
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023
Team	CxServer
Checkmarx Version	8.7.0
Scan Type	Full
Source Origin	LocalPath
Density	6/1000 (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

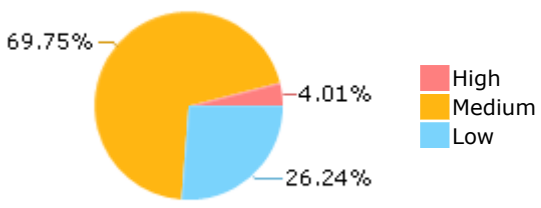
Results Limit

Results limit per query was set to 50

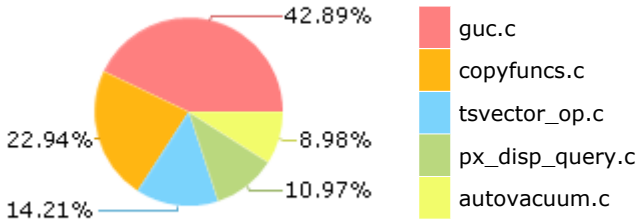
Selected Queries

Selected queries are listed in Result Summary

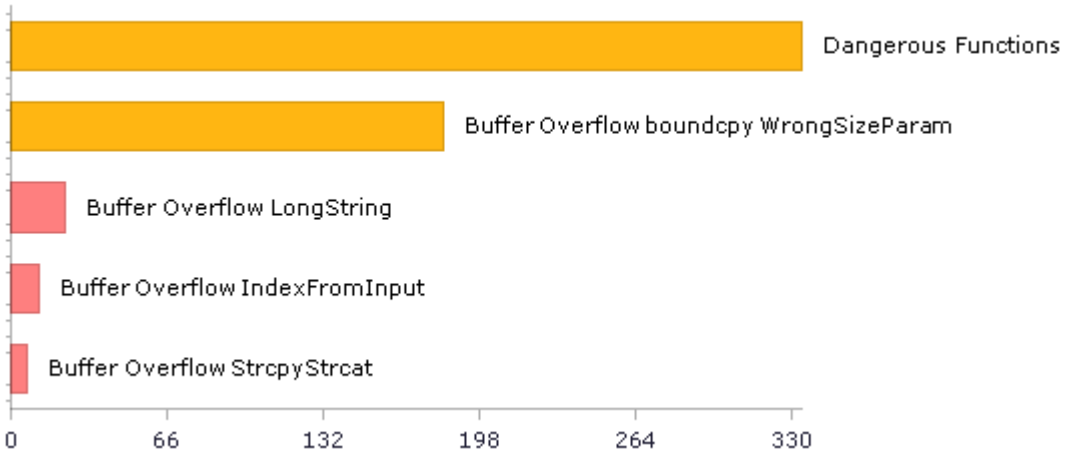
Result Summary



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	279	243
A2-Broken Authentication	App. Specific	EASY	COMMON	AVERAGE	SEVERE	App. Specific	49	49
A3-Sensitive Data Exposure	App. Specific	AVERAGE	WIDESPREAD	AVERAGE	SEVERE	App. Specific	2	2
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	334	334
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	1	1
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	334	334
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	7	7
PCI DSS (3.2) - 6.5.2 - Buffer overflows	220	203
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.	3	3
Audit And Accountability*	Organizations must: (i) create, protect, and retain information system audit records to the extent needed to enable the monitoring, analysis, investigation, and reporting of unlawful, unauthorized, or inappropriate information system activity; and (ii) ensure that the actions of individual information system users can be uniquely traced to those users so they can be held accountable for their actions.	1	1
Configuration Management	Organizations must: (i) establish and maintain baseline configurations and inventories of organizational information systems (including hardware, software, firmware, and documentation) throughout the respective system development life cycles; and (ii) establish and enforce security configuration settings for information technology products employed in organizational information systems.	0	0
Identification And Authentication*	Organizations must identify information system users, processes acting on behalf of users, or devices and authenticate (or verify) the identities of those users, processes, or devices, as a prerequisite to allowing access to organizational information systems.	46	46
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)	49	49
AC-4 Information Flow Enforcement (P1)	0	0
AC-6 Least Privilege (P1)	0	0
AU-9 Protection of Audit Information (P1)	0	0
CM-6 Configuration Settings (P2)	0	0
IA-5 Authenticator Management (P1)	0	0
IA-6 Authenticator Feedback (P2)	0	0
IA-8 Identification and Authentication (Non-Organizational Users) (P1)	0	0
SC-12 Cryptographic Key Establishment and Management (P1)	0	0
SC-13 Cryptographic Protection (P1)	0	0
SC-17 Public Key Infrastructure Certificates (P1)	0	0
SC-18 Mobile Code (P2)	0	0
SC-23 Session Authenticity (P1)*	0	0
SC-28 Protection of Information at Rest (P1)	1	1
SC-4 Information in Shared Resources (P1)	1	1
SC-5 Denial of Service Protection (P1)*	208	112
SC-8 Transmission Confidentiality and Integrity (P1)	0	0
SI-10 Information Input Validation (P1)*	82	62
SI-11 Error Handling (P2)*	114	114
SI-15 Information Output Filtering (P0)	0	0
SI-16 Memory Protection (P1)	9	9

***** 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

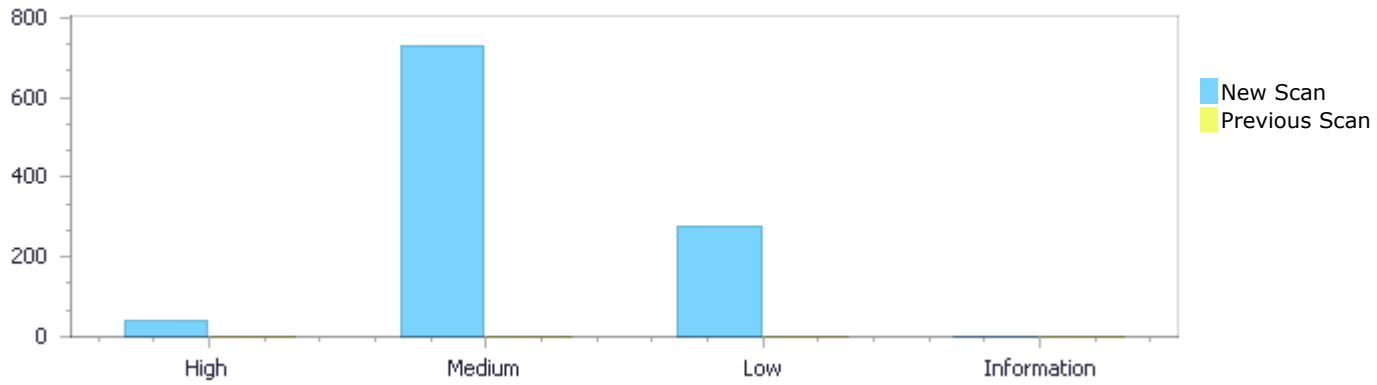
Scan Summary - Custom

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	42	731	275	0	1,048
Recurrent Issues	0	0	0	0	0
Total	42	731	275	0	1,048
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	42	731	275	0	1,048
Urgent	0	0	0	0	0
Proposed Not Exploitable	0	0	0	0	0
Total	42	731	275	0	1,048

Result Summary

Vulnerability Type	Occurrences	Severity
Buffer Overflow LongString	23	High
Buffer Overflow IndexFromInput	12	High
Buffer Overflow StrcpyStrcat	7	High
Dangerous Functions	334	Medium
Buffer Overflow boundcpy WrongSizeParam	183	Medium

Use of Zero Initialized Pointer	130	Medium
Use of Uninitialized Variable	28	Medium
MemoryFree on StackVariable	21	Medium
Use of Uninitialized Pointer	8	Medium
Divide By Zero	5	Medium
Integer Overflow	5	Medium
Memory Leak	4	Medium
Missing Precision	4	Medium
Use After Free	2	Medium
Wrong Size t Allocation	2	Medium
Char Overflow	1	Medium
Double Free	1	Medium
Heap Inspection	1	Medium
Off by One Error in Methods	1	Medium
Uncontrolled Recursion	1	Medium
Unchecked Return Value	114	Low
Improper Resource Access Authorization	46	Low
NULL Pointer Dereference	35	Low
Unchecked Array Index	33	Low
Use of Sizeof On a Pointer Type	22	Low
Potential Precision Problem	9	Low
Potential Off by One Error in Loops	7	Low
Incorrect Permission Assignment For Critical Resources	3	Low
Inconsistent Implementations	2	Low
Arithmenic Operation On Boolean	1	Low
Sizeof Pointer Argument	1	Low
TOCTOU	1	Low
Use of Insufficiently Random Values	1	Low

10 Most Vulnerable Files

High and Medium Vulnerabilities

File Name	Issues Found
PolarDB-for-PostgreSQL-2/guc.c	96
PolarDB-for-PostgreSQL-2/copyfuncs.c	92
PolarDB-for-PostgreSQL-2/tsvector_op.c	52
PolarDB-for-PostgreSQL-2/px_disp_query.c	43
PolarDB-for-PostgreSQL-2/trigger.c	29
PolarDB-for-PostgreSQL-2/execTuples.c	29
PolarDB-for-PostgreSQL-2/autovacuum.c	27
PolarDB-for-PostgreSQL-2/ltree_io.c	24
PolarDB-for-PostgreSQL-2/array_userfuncs.c	23
PolarDB-for-PostgreSQL-2/indexcmds.c	21

Scan Results Details

Buffer Overflow LongString

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

Categories

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

Description

Buffer Overflow LongString\Path 1:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1
Status	New

The size of the buffer used by avlauncher_forkexec in av, at line 365 of PolarDB-for-PostgreSQL-2/autovacuum.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that avlauncher_forkexec passes to "--forkavlauncher", at line 365 of PolarDB-for-PostgreSQL-2/autovacuum.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	371	371
Object	"--forkavlauncher"	av

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method avlauncher_forkexec(void)

```
....
371.        av[ac++] = "--forkavlauncher";
```

Buffer Overflow LongString\Path 2:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=2
Status	New

The size of the buffer used by avworker_forkexec in av, at line 1473 of PolarDB-for-PostgreSQL-2/autovacuum.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that avworker_forkexec passes to "--forkavworker", at line 1473 of PolarDB-for-PostgreSQL-2/autovacuum.c, to overwrite the target buffer.

	Source	Destination

File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	1479	1479
Object	"--forkavworker"	av

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method avworker_forkexec(void)

```
....
1479.       av[ac++] = "--forkavworker";
```

Buffer Overflow LongString\Path 3:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=3
Status	New

The size of the buffer used by pg_event_trigger_ddl_commands in values, at line 2012 of PolarDB-for-PostgreSQL-2/event_trigger.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stringify_adefprivs_objtype passes to "SEQUENCE ", at line 2314 of PolarDB-for-PostgreSQL-2/event_trigger.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/event_trigger.c	PolarDB-for-PostgreSQL-2/event_trigger.c
Line	2323	2182
Object	"SEQUENCE "	values

Code Snippet
File Name PolarDB-for-PostgreSQL-2/event_trigger.c
Method stringify_adefprivs_objtype(ObjectType objtype)

```
....
2323.                   return "SEQUENCES";
```

File Name PolarDB-for-PostgreSQL-2/event_trigger.c
Method pg_event_trigger_ddl_commands(PG_FUNCTION_ARGS)

```
....
2182.                           values[i++] =
CStringGetTextDatum(stringify_adefprivs_objtype(
```

Buffer Overflow LongString\Path 4:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=4
Status	New

The size of the buffer used by pg_event_trigger_ddl_commands in values, at line 2012 of PolarDB-for-PostgreSQL-2/event_trigger.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stringify_adefprivs_objtype passes to "DATABASE ", at line 2314 of PolarDB-for-PostgreSQL-2/event_trigger.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/event_trigger.c	PolarDB-for-PostgreSQL-2/event_trigger.c
Line	2325	2182
Object	"DATABASE "	values

Code Snippet

File Name PolarDB-for-PostgreSQL-2/event_trigger.c

Method stringify_adefprivs_objtype(ObjectType objtype)

```
....
2325.                    return "DATABASES";
```

File Name PolarDB-for-PostgreSQL-2/event_trigger.c

Method pg_event_trigger_ddl_commands(PG_FUNCTION_ARGS)

```
....
2182.                            values[i++] =
CStringGetTextDatum(stringify_adefprivs_objtype(
```

Buffer Overflow LongString\Path 5:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=5
Status	New

The size of the buffer used by pg_event_trigger_ddl_commands in values, at line 2012 of PolarDB-for-PostgreSQL-2/event_trigger.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stringify_adefprivs_objtype passes to "FOREIGN WRAPPER ", at line 2314 of PolarDB-for-PostgreSQL-2/event_trigger.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/event_trigger.c	PolarDB-for-PostgreSQL-2/event_trigger.c
Line	2329	2182
Object	"FOREIGN WRAPPER "	values

Code Snippet

File Name PolarDB-for-PostgreSQL-2/event_trigger.c

Method	stringify_adefprivs_objtype(ObjectType objtype)

```
....
2329.                   return "FOREIGN DATA WRAPPERS";
```

File Name	PolarDB-for-PostgreSQL-2/event_trigger.c
Method	pg_event_trigger_ddl_commands(PG_FUNCTION_ARGS)

```
....
2182.                           values[i++] =
CStringGetTextDatum(stringify_adefprivs_objtype(
```

Buffer Overflow LongString\Path 6:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=6
Status	New

The size of the buffer used by pg_event_trigger_ddl_commands in values, at line 2012 of PolarDB-for-PostgreSQL-2/event_trigger.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stringify_adefprivs_objtype passes to "FOREIGN SERVER ", at line 2314 of PolarDB-for-PostgreSQL-2/event_trigger.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/event_trigger.c	PolarDB-for-PostgreSQL-2/event_trigger.c
Line	2331	2182
Object	"FOREIGN SERVER "	values

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/event_trigger.c
Method	stringify_adefprivs_objtype(ObjectType objtype)

```
....
2331.                   return "FOREIGN SERVERS";
```

File Name	PolarDB-for-PostgreSQL-2/event_trigger.c
Method	pg_event_trigger_ddl_commands(PG_FUNCTION_ARGS)

```
....
2182.                           values[i++] =
CStringGetTextDatum(stringify_adefprivs_objtype(
```

Buffer Overflow LongString\Path 7:

Severity	High
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=7
Status	New

The size of the buffer used by pg_event_trigger_ddl_commands in values, at line 2012 of PolarDB-for-PostgreSQL-2/event_trigger.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stringify_adefprivs_objtype passes to "FUNCTION ", at line 2314 of PolarDB-for-PostgreSQL-2/event_trigger.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/event_trigger.c	PolarDB-for-PostgreSQL-2/event_trigger.c
Line	2333	2182
Object	"FUNCTION "	values

Code Snippet

File Name PolarDB-for-PostgreSQL-2/event_trigger.c

Method stringify_adefprivs_objtype(ObjectType objtype)

```
....
2333.                   return "FUNCTIONS";
```

File Name PolarDB-for-PostgreSQL-2/event_trigger.c

Method pg_event_trigger_ddl_commands(PG_FUNCTION_ARGS)

```
....
2182.                           values[i++] =
CStringGetTextDatum(stringify_adefprivs_objtype(
```

Buffer Overflow LongString\Path 8:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=8
Status	New

The size of the buffer used by pg_event_trigger_ddl_commands in values, at line 2012 of PolarDB-for-PostgreSQL-2/event_trigger.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stringify_adefprivs_objtype passes to "LANGUAGE ", at line 2314 of PolarDB-for-PostgreSQL-2/event_trigger.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/event_trigger.c	PolarDB-for-PostgreSQL-2/event_trigger.c
Line	2335	2182
Object	"LANGUAGE "	values

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/event_trigger.c
Method	stringify_adefprivs_objtype(ObjectType objtype)

```
....
2335.                    return "LANGUAGES";
```

File Name	PolarDB-for-PostgreSQL-2/event_trigger.c
Method	pg_event_trigger_ddl_commands(PG_FUNCTION_ARGS)

```
....
2182.                            values[i++] =
CStringGetTextDatum(stringify_adefprivs_objtype(
```

Buffer Overflow LongString\Path 9:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=9
Status	New

The size of the buffer used by pg_event_trigger_ddl_commands in values, at line 2012 of PolarDB-for-PostgreSQL-2/event_trigger.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stringify_adefprivs_objtype passes to "LARGE OBJECT ", at line 2314 of PolarDB-for-PostgreSQL-2/event_trigger.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/event_trigger.c	PolarDB-for-PostgreSQL-2/event_trigger.c
Line	2337	2182
Object	"LARGE OBJECT "	values

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/event_trigger.c
Method	stringify_adefprivs_objtype(ObjectType objtype)

```
....
2337.                    return "LARGE OBJECTS";
```

File Name	PolarDB-for-PostgreSQL-2/event_trigger.c
Method	pg_event_trigger_ddl_commands(PG_FUNCTION_ARGS)

```
....
2182.                            values[i++] =
CStringGetTextDatum(stringify_adefprivs_objtype(
```

Buffer Overflow LongString\Path 10:

Severity	High

Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=10
Status	New

The size of the buffer used by pg_event_trigger_ddl_commands in values, at line 2012 of PolarDB-for-PostgreSQL-2/event_trigger.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stringify_adefprivs_objtype passes to "PROCEDURE ", at line 2314 of PolarDB-for-PostgreSQL-2/event_trigger.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/event_trigger.c	PolarDB-for-PostgreSQL-2/event_trigger.c
Line	2341	2182
Object	"PROCEDURE "	values

Code Snippet

File Name PolarDB-for-PostgreSQL-2/event_trigger.c

Method stringify_adefprivs_objtype(ObjectType objtype)

```
....
2341.                   return "PROCEDURES";
```

File Name PolarDB-for-PostgreSQL-2/event_trigger.c

Method pg_event_trigger_ddl_commands(PG_FUNCTION_ARGS)

```
....
2182.                           values[i++] =
CStringGetTextDatum(stringify_adefprivs_objtype(
```

Buffer Overflow LongString\Path 11:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=11
Status	New

The size of the buffer used by pg_event_trigger_ddl_commands in values, at line 2012 of PolarDB-for-PostgreSQL-2/event_trigger.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that stringify_adefprivs_objtype passes to "TABLESPACE ", at line 2314 of PolarDB-for-PostgreSQL-2/event_trigger.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/event_trigger.c	PolarDB-for-PostgreSQL-2/event_trigger.c
Line	2345	2182
Object	"TABLESPACE "	values

Code Snippet
File Name PolarDB-for-PostgreSQL-2/event_trigger.c
Method stringify_adefprivs_objtype(ObjectType objtype)

```
....
2345.                   return "TABLESPACES";
```

File Name PolarDB-for-PostgreSQL-2/event_trigger.c

Method pg_event_trigger_ddl_commands(PG_FUNCTION_ARGS)

```
....
2182.                           values[i++] =
CStringGetTextDatum(stringify_adefprivs_objtype(
```

Buffer Overflow LongString\Path 12:

Severity High
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=12
Status New

The size of the buffer used by main in ts, at line 106 of PolarDB-for-PostgreSQL-2/sql-array.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to "2000-1-1 0%d:00:00", at line 106 of PolarDB-for-PostgreSQL-2/sql-array.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/sql-array.c	PolarDB-for-PostgreSQL-2/sql-array.c
Line	163	164
Object	"2000-1-1 0%d:00:00"	ts

Code Snippet
File Name PolarDB-for-PostgreSQL-2/sql-array.c
Method main (void)

```
....
163.              sprintf(str, "2000-1-1 0%d:00:00", j);
164.              ts[j] = PGTYPEStimestamp_from_asc(str, NULL);
```

Buffer Overflow LongString\Path 13:

Severity High
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=13
Status New

The size of the buffer used by main in d, at line 106 of PolarDB-for-PostgreSQL-2/sql-array.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that

main passes to "2000-1-1 0%d:00:00", at line 106 of PolarDB-for-PostgreSQL-2/sql-array.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/sql-array.c	PolarDB-for-PostgreSQL-2/sql-array.c
Line	163	166
Object	"2000-1-1 0%d:00:00"	d

Code Snippet
File Name PolarDB-for-PostgreSQL-2/sql-array.c
Method main (void)

```
....
163.              sprintf(str, "2000-1-1 0%d:00:00", j);
....
166.              d[j] = PGTYPESdate_from_asc(str, NULL);
```

Buffer Overflow LongString\Path 14:

Severity High
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=14
Status New

The size of the buffer used by main in in, at line 106 of PolarDB-for-PostgreSQL-2/sql-array.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to "2000-1-1 0%d:00:00", at line 106 of PolarDB-for-PostgreSQL-2/sql-array.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/sql-array.c	PolarDB-for-PostgreSQL-2/sql-array.c
Line	163	169
Object	"2000-1-1 0%d:00:00"	in

Code Snippet
File Name PolarDB-for-PostgreSQL-2/sql-array.c
Method main (void)

```
....
163.              sprintf(str, "2000-1-1 0%d:00:00", j);
....
169.              in[j] = *inter;
```

Buffer Overflow LongString\Path 15:

Severity High
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=15
Status New

The size of the buffer used by main in d, at line 106 of PolarDB-for-PostgreSQL-2/sql-array.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to "2000-1-1%d\n", at line 106 of PolarDB-for-PostgreSQL-2/sql-array.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/sql-array.c	PolarDB-for-PostgreSQL-2/sql-array.c
Line	165	166
Object	"2000-1-1%d\n"	d

Code Snippet
File Name PolarDB-for-PostgreSQL-2/sql-array.c
Method main (void)

```
....
165.                sprintf(str, "2000-1-1%d\n", j);
166.                d[j] = PGTYPESdate_from_asc(str, NULL);
```

Buffer Overflow LongString\Path 16:

Severity High
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=16
Status New

The size of the buffer used by main in in, at line 106 of PolarDB-for-PostgreSQL-2/sql-array.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that main passes to "2000-1-1%d\n", at line 106 of PolarDB-for-PostgreSQL-2/sql-array.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/sql-array.c	PolarDB-for-PostgreSQL-2/sql-array.c
Line	165	169
Object	"2000-1-1%d\n"	in

Code Snippet
File Name PolarDB-for-PostgreSQL-2/sql-array.c
Method main (void)

```
....
165.                sprintf(str, "2000-1-1%d\n", j);
....
169.                in[j] = *inter;
```

Buffer Overflow LongString\Path 17:

Severity High
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=200

	23&pathid=17
Status	New

The size of the buffer used by dttofmtasc_replace in tmp, at line 310 of PolarDB-for-PostgreSQL-2/timestamp.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that dttofmtasc_replace passes to "%H:%M:% ", at line 310 of PolarDB-for-PostgreSQL-2/timestamp.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/timestamp.c	PolarDB-for-PostgreSQL-2/timestamp.c
Line	574	404
Object	"%H:%M:% "	tmp

Code Snippet
File Name PolarDB-for-PostgreSQL-2/timestamp.c
Method dttofmtasc_replace(timestamp * ts, date dDate, int dow, struct tm *tm,

```
....
574.
"%H:%M:%S");
....
404.                                        tmp[2] = *p;
```

Buffer Overflow LongString\Path 18:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=18
Status	New

The size of the buffer used by dttofmtasc_replace in tmp, at line 310 of PolarDB-for-PostgreSQL-2/timestamp.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that dttofmtasc_replace passes to "%H:%M", at line 310 of PolarDB-for-PostgreSQL-2/timestamp.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/timestamp.c	PolarDB-for-PostgreSQL-2/timestamp.c
Line	551	404
Object	"%H:%M"	tmp

Code Snippet
File Name PolarDB-for-PostgreSQL-2/timestamp.c
Method dttofmtasc_replace(timestamp * ts, date dDate, int dow, struct tm *tm,

```
....
551.
"%H:%M");
....
404.                                        tmp[2] = *p;
```

Buffer Overflow LongString\Path 19:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=19
Status	New

The size of the buffer used by dttofmtasc_replace in tmp, at line 310 of PolarDB-for-PostgreSQL-2/timestamp.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that dttofmtasc_replace passes to "%I:%M:%S %p", at line 310 of PolarDB-for-PostgreSQL-2/timestamp.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/timestamp.c	PolarDB-for-PostgreSQL-2/timestamp.c
Line	543	404
Object	"%I:%M:%S %p"	tmp

Code Snippet
File Name PolarDB-for-PostgreSQL-2/timestamp.c
Method dttofmtasc_replace(timestamp * ts, date dDate, int dow, struct tm *tm,

```
....
543.
"%I:%M:%S %p");
....
404.                                        tmp[2] = *p;
```

Buffer Overflow LongString\Path 20:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=20
Status	New

The size of the buffer used by dttofmtasc_replace in tmp, at line 310 of PolarDB-for-PostgreSQL-2/timestamp.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that dttofmtasc_replace passes to "%m/%d/%y", at line 310 of PolarDB-for-PostgreSQL-2/timestamp.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/timestamp.c	PolarDB-for-PostgreSQL-2/timestamp.c
Line	384	404
Object	"%m/%d/%y"	tmp

Code Snippet
File Name PolarDB-for-PostgreSQL-2/timestamp.c
Method dttofmtasc_replace(timestamp * ts, date dDate, int dow, struct tm *tm,

```
....
384.
"%m/%d/%y");
....
404.                                        tmp[2] = *p;
```

Buffer Overflow LongString\Path 21:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=21
Status	New

The size of the buffer used by IdentifySystem in values, at line 394 of PolarDB-for-PostgreSQL-2/walsender.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that IdentifySystem passes to "%X/%X", at line 394 of PolarDB-for-PostgreSQL-2/walsender.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/walsender.c	PolarDB-for-PostgreSQL-2/walsender.c
Line	431	471
Object	"%X/%X"	values

Code Snippet
File Name PolarDB-for-PostgreSQL-2/walsender.c
Method IdentifySystem(void)

```
....
431.        snprintf(xloc, sizeof(xloc), "%X/%X", (uint32) (logptr >>
32), (uint32) logptr);
....
471.        values[2] = CStringGetTextDatum(xloc);
```

Buffer Overflow LongString\Path 22:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=22
Status	New

The size of the buffer used by StartReplication in values, at line 588 of PolarDB-for-PostgreSQL-2/walsender.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that StartReplication passes to "%X/%X", at line 588 of PolarDB-for-PostgreSQL-2/walsender.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/walsender.c	PolarDB-for-PostgreSQL-2/walsender.c
Line	881	903
Object	"%X/%X"	values

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/walsender.c
Method	StartReplication(StartReplicationCmd *cmd)

```
....
881.              snprintf(startpos_str, sizeof(startpos_str), "%X/%X",
....
903.              values[1] = CStringGetTextDatum(startpos_str);
```

Buffer Overflow LongString\Path 23:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=23
Status	New

The size of the buffer used by CreateReplicationSlot in values, at line 1010 of PolarDB-for-PostgreSQL-2/walsender.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that CreateReplicationSlot passes to "%X/%X", at line 1010 of PolarDB-for-PostgreSQL-2/walsender.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/walsender.c	PolarDB-for-PostgreSQL-2/walsender.c
Line	1149	1182
Object	"%X/%X"	values

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/walsender.c
Method	CreateReplicationSlot(CreateReplicationSlotCmd *cmd)

```
....
1149.       snprintf(xloc, sizeof(xloc), "%X/%X",
....
1182.       values[1] = CStringGetTextDatum(xloc);
```

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=1020028&projectid=20023&pathid=31
Status	New

The size of the buffer used by apw_load_buffers in i, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that apw_load_buffers passes to file, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	324	349
Object	file	i

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
324.        if (fscanf(file, "<<%d>>\n", &num_elements) != 1)
....
349.                                &forknum, &blkinfo[i].blocknum) != 5)
```

Buffer Overflow IndexFromInput\Path 2:

Severity High
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=32
Status New

The size of the buffer used by apw_load_buffers in i, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that apw_load_buffers passes to Address, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	324	349
Object	Address	i

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
324.        if (fscanf(file, "<<%d>>\n", &num_elements) != 1)
....
349.                                &forknum, &blkinfo[i].blocknum) != 5)
```

Buffer Overflow IndexFromInput\Path 3:

Severity High
Result State To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=33
Status	New

The size of the buffer used by apw_load_buffers in i, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that apw_load_buffers passes to file, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	324	348
Object	file	i

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
324.        if (fscanf(file, "<<%d>>\n", &num_elements) != 1)
....
348.                                &blkinfo[i].tablespace,
&blkinfo[i].filenode,
```

Buffer Overflow IndexFromInput\Path 4:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=34
Status	New

The size of the buffer used by apw_load_buffers in i, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that apw_load_buffers passes to Address, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	324	348
Object	Address	i

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
324.        if (fscanf(file, "<<%d>>\n", &num_elements) != 1)
....
348.                                &blkinfo[i].tablespace,
&blkinfo[i].filenode,
```

Buffer Overflow IndexFromInput\Path 5:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=35
Status	New

The size of the buffer used by apw_load_buffers in i, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that apw_load_buffers passes to file, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	324	348
Object	file	i

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
324.        if (fscanf(file, "<<%d>>\n", &num_elements) != 1)
....
348.                                &blkinfo[i].tablespace,
&blkinfo[i].filenode,
```

Buffer Overflow IndexFromInput\Path 6:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=36
Status	New

The size of the buffer used by apw_load_buffers in i, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that apw_load_buffers passes to Address, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c

Line	324	348
Object	Address	i

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
324.        if (fscanf(file, "<<%d>>\n", &num_elements) != 1)
....
348.                                &blkinfo[i].tablespace,
&blkinfo[i].filenode,
```

Buffer Overflow IndexFromInput\Path 7:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=37
Status	New

The size of the buffer used by apw_load_buffers in i, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that apw_load_buffers passes to file, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	324	347
Object	file	i

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
324.        if (fscanf(file, "<<%d>>\n", &num_elements) != 1)
....
347.              if (fscanf(file, "%u,%u,%u,%u,%u\n",
&blkinfo[i].database,
```

Buffer Overflow IndexFromInput\Path 8:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=38
Status	New

The size of the buffer used by apw_load_buffers in i, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the

source buffer that apw_load_buffers passes to Address, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	324	347
Object	Address	i

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
324.        if (fscanf(file, "<<%d>>\n", &num_elements) != 1)
....
347.              if (fscanf(file, "%u,%u,%u,%u,%u\n",
&blkinfo[i].database,
```

Buffer Overflow IndexFromInput\Path 9:

Severity High
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=39
Status New

The size of the buffer used by apw_load_buffers in i, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that apw_load_buffers passes to Address, at line 275 of PolarDB-for-PostgreSQL-2/autoprewarm.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	349	353
Object	Address	i

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
349.                              &forknum, &blkinfo[i].blocknum) != 5)
....
353.              blkinfo[i].forknum = forknum;
```

Buffer Overflow IndexFromInput\Path 10:

Severity High
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=200

	23&pathid=40
Status	New

The size of the buffer used by StartupReplicationOrigin in last_state, at line 682 of PolarDB-for-PostgreSQL-2/origin.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that StartupReplicationOrigin passes to Address, at line 682 of PolarDB-for-PostgreSQL-2/origin.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/origin.c	PolarDB-for-PostgreSQL-2/origin.c
Line	741	776
Object	Address	last_state

Code Snippet
File Name PolarDB-for-PostgreSQL-2/origin.c
Method StartupReplicationOrigin(void)

```
....
741.                readBytes = read(fd, &disk_state, sizeof(disk_state));
....
776.                replication_states[last_state].remote_lsn =
disk_state.remote_lsn;
```

Buffer Overflow IndexFromInput\Path 11:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=41
Status	New

The size of the buffer used by StartupReplicationOrigin in last_state, at line 682 of PolarDB-for-PostgreSQL-2/origin.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that StartupReplicationOrigin passes to Address, at line 682 of PolarDB-for-PostgreSQL-2/origin.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/origin.c	PolarDB-for-PostgreSQL-2/origin.c
Line	741	775
Object	Address	last_state

Code Snippet
File Name PolarDB-for-PostgreSQL-2/origin.c
Method StartupReplicationOrigin(void)

```
....
741.                readBytes = read(fd, &disk_state, sizeof(disk_state));
....
775.                replication_states[last_state].roident =
disk_state.roident;
```

Buffer Overflow IndexFromInput\Path 12:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=42
Status	New

The size of the buffer used by pg_tablespace_location in rllen, at line 506 of PolarDB-for-PostgreSQL-2/misc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that pg_tablespace_location passes to targetpath, at line 506 of PolarDB-for-PostgreSQL-2/misc.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/misc.c	PolarDB-for-PostgreSQL-2/misc.c
Line	543	554
Object	targetpath	rllen

Code Snippet
File Name PolarDB-for-PostgreSQL-2/misc.c
Method pg_tablespace_location(PG_FUNCTION_ARGS)

```
....
543.        rllen = readlink(sourcepath, targetpath,
sizeof(targetpath));
....
554.        targetpath[rllen] = '\0';
```

Buffer Overflow StrcpyStrcat

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

Categories

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

Description
Buffer Overflow StrcpyStrcat\Path 1:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=24
Status	New

The size of the buffer used by makeObjectName in ndx, at line 2058 of PolarDB-for-PostgreSQL-2/indexcmds.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that makeObjectName passes to name2, at line 2058 of PolarDB-for-PostgreSQL-2/indexcmds.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/indexcmds.c	PolarDB-for-PostgreSQL-2/indexcmds.c
Line	2058	2111

Object	name2	ndx

Code Snippet
File Name PolarDB-for-PostgreSQL-2/indexcmds.c
Method makeObjectName(const char *name1, const char *name2, const char *label)

```
....
2058.  makeObjectName(const char *name1, const char *name2, const char
*label)
....
2111.              strcpy(name + ndx, label);
```

Buffer Overflow StrcpyStrcat\Path 2:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=25
Status	New

The size of the buffer used by makeObjectName in BinaryExpr, at line 2058 of PolarDB-for-PostgreSQL-2/indexcmds.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that makeObjectName passes to name2, at line 2058 of PolarDB-for-PostgreSQL-2/indexcmds.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/indexcmds.c	PolarDB-for-PostgreSQL-2/indexcmds.c
Line	2058	2111
Object	name2	BinaryExpr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/indexcmds.c
Method makeObjectName(const char *name1, const char *name2, const char *label)

```
....
2058.  makeObjectName(const char *name1, const char *name2, const char
*label)
....
2111.              strcpy(name + ndx, label);
```

Buffer Overflow StrcpyStrcat\Path 3:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=26
Status	New

The size of the buffer used by makeObjectName in label, at line 2058 of PolarDB-for-PostgreSQL-2/indexcmds.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that makeObjectName passes to label, at line 2058 of PolarDB-for-PostgreSQL-2/indexcmds.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/indexcmds.c	PolarDB-for-PostgreSQL-2/indexcmds.c
Line	2058	2111
Object	label	label

Code Snippet
File Name PolarDB-for-PostgreSQL-2/indexcmds.c
Method makeObjectName(const char *name1, const char *name2, const char *label)

```
....
2058.  makeObjectName(const char *name1, const char *name2, const char
*label)
....
2111.              strcpy(name + ndx, label);
```

Buffer Overflow StrcpyStrcat\Path 4:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=27
Status	New

The size of the buffer used by EncodeSpecialTimestamp in str, at line 196 of PolarDB-for-PostgreSQL-2/timestamp.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that EncodeSpecialTimestamp passes to str, at line 196 of PolarDB-for-PostgreSQL-2/timestamp.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/timestamp.c	PolarDB-for-PostgreSQL-2/timestamp.c
Line	196	199
Object	str	str

Code Snippet
File Name PolarDB-for-PostgreSQL-2/timestamp.c
Method EncodeSpecialTimestamp(timestamp dt, char *str)

```
....
196.  EncodeSpecialTimestamp(timestamp dt, char *str)
....
199.              strcpy(str, EARLY);
```

Buffer Overflow StrcpyStrcat\Path 5:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=28
Status	New

The size of the buffer used by EncodeSpecialTimestamp in str, at line 196 of PolarDB-for-PostgreSQL-2/timestamp.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that EncodeSpecialTimestamp passes to str, at line 196 of PolarDB-for-PostgreSQL-2/timestamp.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/timestamp.c	PolarDB-for-PostgreSQL-2/timestamp.c
Line	196	201
Object	str	str

Code Snippet
File Name PolarDB-for-PostgreSQL-2/timestamp.c
Method EncodeSpecialTimestamp(timestamp dt, char *str)

```
....
196.  EncodeSpecialTimestamp(timestamp dt, char *str)
....
201.              strcpy(str, LATE);
```

Buffer Overflow StrcpyStrcat\Path 6:

Severity	High
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=29
Status	New

The size of the buffer used by standard_ProcessUtility in completionTag, at line 494 of PolarDB-for-PostgreSQL-2/utility.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that standard_ProcessUtility passes to completionTag, at line 494 of PolarDB-for-PostgreSQL-2/utility.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/utility.c	PolarDB-for-PostgreSQL-2/utility.c
Line	500	564
Object	completionTag	completionTag

Code Snippet
File Name PolarDB-for-PostgreSQL-2/utility.c
Method standard_ProcessUtility(PlannedStmt *pstmt,

```
....
500.                                    char *completionTag)
....
564.
      strcpy(completionTag, "ROLLBACK");
```

Buffer Overflow StrcpyStrcat\Path 7:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=30
Status	New

The size of the buffer used by standard_ProcessUtility in completionTag, at line 494 of PolarDB-for-PostgreSQL-2/utility.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that standard_ProcessUtility passes to completionTag, at line 494 of PolarDB-for-PostgreSQL-2/utility.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/utility.c	PolarDB-for-PostgreSQL-2/utility.c
Line	500	574
Object	completionTag	completionTag

Code Snippet
File Name PolarDB-for-PostgreSQL-2/utility.c
Method standard_ProcessUtility(PlannedStmt *pstmt,

```
....
500.                                        char *completionTag)
....
574.
      strcpy(completionTag, "ROLLBACK");
```

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=1020028&projectid=20023&pathid=284
Status	New

The dangerous function, memcpy, was found in use at line 200 in PolarDB-for-PostgreSQL-2/_ltree_op.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/_ltree_op.c	PolarDB-for-PostgreSQL-2/_ltree_op.c
Line	215	215
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/_ltree_op.c

Method	_ltree_extract_isparent(PG_FUNCTION_ARGS)

```
....
215.        memcpy(item, found, VARSIZE(found));
```

Dangerous Functions\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=285
Status	New

The dangerous function, memcpy, was found in use at line 223 in PolarDB-for-PostgreSQL-2/_ltree_op.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/_ltree_op.c	PolarDB-for-PostgreSQL-2/_ltree_op.c
Line	238	238
Object	memcpy	memcpy

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/_ltree_op.c
Method	_ltree_extract_risparent(PG_FUNCTION_ARGS)

```
....
238.        memcpy(item, found, VARSIZE(found));
```

Dangerous Functions\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=286
Status	New

The dangerous function, memcpy, was found in use at line 246 in PolarDB-for-PostgreSQL-2/_ltree_op.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/_ltree_op.c	PolarDB-for-PostgreSQL-2/_ltree_op.c
Line	261	261
Object	memcpy	memcpy

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/_ltree_op.c
Method	_ltq_extract_regex(PG_FUNCTION_ARGS)

```
....
261.        memcpy(item, found, VARSIZE(found));
```

Dangerous Functions\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=287
Status	New

The dangerous function, memcpy, was found in use at line 269 in PolarDB-for-PostgreSQL-2/_ltree_op.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/_ltree_op.c	PolarDB-for-PostgreSQL-2/_ltree_op.c
Line	284	284
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/_ltree_op.c
Method _ltxtq_extract_exec(PG_FUNCTION_ARGS)

```
....
284.        memcpy(item, found, VARSIZE(found));
```

Dangerous Functions\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=288
Status	New

The dangerous function, memcpy, was found in use at line 218 in PolarDB-for-PostgreSQL-2/array_userfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	366	366
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method array_cat(PG_FUNCTION_ARGS)

```
....
366.                memcpy(dims, dims2, ndims * sizeof(int));
```

Dangerous Functions\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=289
Status	New

The dangerous function, memcpy, was found in use at line 218 in PolarDB-for-PostgreSQL-2/array_userfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	367	367
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method array_cat(PG_FUNCTION_ARGS)

```
....
367.                memcpy(lbs, lbs2, ndims * sizeof(int));
```

Dangerous Functions\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=290
Status	New

The dangerous function, memcpy, was found in use at line 218 in PolarDB-for-PostgreSQL-2/array_userfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	394	394
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/array_userfuncs.c

Method	array_cat(PG_FUNCTION_ARGS)

```
....
394.                memcpy(dims, dims1, ndims * sizeof(int));
```

Dangerous Functions\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=291
Status	New

The dangerous function, memcpy, was found in use at line 218 in PolarDB-for-PostgreSQL-2/array_userfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	395	395
Object	memcpy	memcpy

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method	array_cat(PG_FUNCTION_ARGS)

```
....
395.                memcpy(lbs, lbs1, ndims * sizeof(int));
```

Dangerous Functions\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=292
Status	New

The dangerous function, memcpy, was found in use at line 218 in PolarDB-for-PostgreSQL-2/array_userfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	433	433
Object	memcpy	memcpy

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method	array_cat(PG_FUNCTION_ARGS)

```
....
433.        memcpy(ARR_DIMS(result), dims, ndims * sizeof(int));
```

Dangerous Functions\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=293
Status	New

The dangerous function, memcpy, was found in use at line 218 in PolarDB-for-PostgreSQL-2/array_userfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	434	434
Object	memcpy	memcpy

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method	array_cat(PG_FUNCTION_ARGS)

```
....
434.        memcpy(ARR_LBOUND(result), lbs, ndims * sizeof(int));
```

Dangerous Functions\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=294
Status	New

The dangerous function, memcpy, was found in use at line 218 in PolarDB-for-PostgreSQL-2/array_userfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	436	436
Object	memcpy	memcpy

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method	array_cat(PG_FUNCTION_ARGS)

```
....
436.        memcpy(ARR_DATA_PTR(result), dat1, ndatabytes1);
```

Dangerous Functions\Path 12:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=295
Status	New

The dangerous function, memcpy, was found in use at line 218 in PolarDB-for-PostgreSQL-2/array_userfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	437	437
Object	memcpy	memcpy

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method	array_cat(PG_FUNCTION_ARGS)

```
....
437.        memcpy(ARR_DATA_PTR(result) + ndatabytes1, dat2,
ndatabytes2);
```

Dangerous Functions\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=296
Status	New

The dangerous function, memcpy, was found in use at line 952 in PolarDB-for-PostgreSQL-2/autovacuum.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	1088	1088
Object	memcpy	memcpy

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/autovacuum.c
Method	rebuild_database_list(Oid newdb)

```
....
1088.                    memcpy(&(dbary[i++]), db, sizeof(avl_dbase));
```

Dangerous Functions\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=297
Status	New

The dangerous function, memcpy, was found in use at line 1968 in PolarDB-for-PostgreSQL-2/autovacuum.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	2167	2167
Object	memcpy	memcpy

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/autovacuum.c
Method	do_autovacuum(void)

```
....
2167.                                memcpy(&hentry->ar_reloptions,
relopts,
```

Dangerous Functions\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=298
Status	New

The dangerous function, memcpy, was found in use at line 2757 in PolarDB-for-PostgreSQL-2/autovacuum.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	2771	2771
Object	memcpy	memcpy

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/autovacuum.c

Method	extract_autovac_opts(HeapTuple tup, TupleDesc pg_class_desc)

```
....
2771.       memcpy(av, &(((StdRdOptions *) relopts)->autovacuum),
sizeof(AutoVacOpts));
```

Dangerous Functions\Path 16:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=299
Status	New

The dangerous function, memcpy, was found in use at line 1110 in PolarDB-for-PostgreSQL-2/be-secure-openssl.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/be-secure-openssl.c	PolarDB-for-PostgreSQL-2/be-secure-openssl.c
Line	1157	1157
Object	memcpy	memcpy

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/be-secure-openssl.c
Method	be_tls_get_certificate_hash(Port *port, size_t *len)

```
....
1157.       memcpy(cert_hash, hash, hash_size);
```

Dangerous Functions\Path 17:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=300
Status	New

The dangerous function, memcpy, was found in use at line 310 in PolarDB-for-PostgreSQL-2/btreefuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/btreefuncs.c	PolarDB-for-PostgreSQL-2/btreefuncs.c
Line	369	369
Object	memcpy	memcpy

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/btreefuncs.c

Method	bt_page_items(PG_FUNCTION_ARGS)

```
....
369.                memcpy(uargs->page, BufferGetPage(buffer), BLCKSZ);
```

Dangerous Functions\Path 18:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=301
Status	New

The dangerous function, memcpy, was found in use at line 80 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	118	118
Object	memcpy	memcpy

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/copyfuncs.c
Method	_copyPlannedStmt(const PlannedStmt *from)

```
....
118.                COPY_POINTER_FIELD(subplan_sliceIds, list_length(from->subplans) * sizeof(int));
```

Dangerous Functions\Path 19:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=302
Status	New

The dangerous function, memcpy, was found in use at line 184 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	202	202
Object	memcpy	memcpy

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/copyfuncs.c
Method	_copyResult(const Result *from)

```
....
202.              COPY_POINTER_FIELD(hashFilterColIdx, from-
>numHashFilterCols * sizeof(AttrNumber));
```

Dangerous Functions\Path 20:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=303
Status	New

The dangerous function, memcpy, was found in use at line 184 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	203	203
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyResult(const Result *from)

```
....
203.              COPY_POINTER_FIELD(hashFilterFuncs, from-
>numHashFilterCols * sizeof(Oid));
```

Dangerous Functions\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=304
Status	New

The dangerous function, memcpy, was found in use at line 311 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	326	326
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyMergeAppend(const MergeAppend *from)

```
....
326.        COPY_POINTER_FIELD(sortColIdx, from->numCols *
sizeof(AttrNumber));
```

Dangerous Functions\Path 22:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=305
Status	New

The dangerous function, memcpy, was found in use at line 311 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	327	327
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyMergeAppend(const MergeAppend *from)

```
....
327.        COPY_POINTER_FIELD(sortOperators, from->numCols *
sizeof(Oid));
```

Dangerous Functions\Path 23:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=306
Status	New

The dangerous function, memcpy, was found in use at line 311 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	328	328
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyMergeAppend(const MergeAppend *from)

```
....
328.        COPY_POINTER_FIELD(collations, from->numCols * sizeof(Oid));
```

Dangerous Functions\Path 24:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=307
Status	New

The dangerous function, memcpy, was found in use at line 311 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	329	329
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyMergeAppend(const MergeAppend *from)

```
....
329.        COPY_POINTER_FIELD(nullsFirst, from->numCols *
sizeof(bool));
```

Dangerous Functions\Path 25:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=308
Status	New

The dangerous function, memcpy, was found in use at line 338 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	354	354
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyRecursiveUnion(const RecursiveUnion *from)

```
....
354.                COPY_POINTER_FIELD(dupColIdx, from->numCols *
sizeof(AttrNumber));
```

Dangerous Functions\Path 26:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=309
Status	New

The dangerous function, memcpy, was found in use at line 338 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	355	355
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyRecursiveUnion(const RecursiveUnion *from)

```
....
355.                COPY_POINTER_FIELD(dupOperators, from->numCols *
sizeof(Oid));
```

Dangerous Functions\Path 27:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=310
Status	New

The dangerous function, memcpy, was found in use at line 434 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	449	449
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyGatherMerge(const GatherMerge *from)

```
....
449.        COPY_POINTER_FIELD(sortColIdx, from->numCols *
sizeof(AttrNumber));
```

Dangerous Functions\Path 28:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=311
Status	New

The dangerous function, memcpy, was found in use at line 434 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	450	450
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyGatherMerge(const GatherMerge *from)

```
....
450.        COPY_POINTER_FIELD(sortOperators, from->numCols *
sizeof(Oid));
```

Dangerous Functions\Path 29:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=312
Status	New

The dangerous function, memcpy, was found in use at line 434 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	451	451
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyGatherMerge(const GatherMerge *from)

```
....
451.        COPY_POINTER_FIELD(collations, from->numCols * sizeof(Oid));
```

Dangerous Functions\Path 30:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=313
Status	New

The dangerous function, memcpy, was found in use at line 434 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	452	452
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyGatherMerge(const GatherMerge *from)

```
....
452.        COPY_POINTER_FIELD(nullsFirst, from->numCols *
sizeof(bool));
```

Dangerous Functions\Path 31:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=314
Status	New

The dangerous function, memcpy, was found in use at line 916 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	934	934
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyMergeJoin(const MergeJoin *from)

```
....
934.                COPY_POINTER_FIELD(mergeFamilies, numCols *
sizeof(Oid));
```

Dangerous Functions\Path 32:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=315
Status	New

The dangerous function, memcpy, was found in use at line 916 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	935	935
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyMergeJoin(const MergeJoin *from)

```
....
935.                COPY_POINTER_FIELD(mergeCollations, numCols *
sizeof(Oid));
```

Dangerous Functions\Path 33:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=316
Status	New

The dangerous function, memcpy, was found in use at line 916 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	936	936
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyMergeJoin(const MergeJoin *from)

```
....
936.                COPY_POINTER_FIELD(mergeStrategies, numCols *
sizeof(int));
```

Dangerous Functions\Path 34:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=317
Status	New

The dangerous function, memcpy, was found in use at line 916 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	937	937
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyMergeJoin(const MergeJoin *from)

```
....
937.                COPY_POINTER_FIELD(mergeNullsFirst, numCols *
sizeof(bool));
```

Dangerous Functions\Path 35:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=318
Status	New

The dangerous function, memcpy, was found in use at line 1007 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1017	1017
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copySort(const Sort *from)

```
....
1017.       COPY_POINTER_FIELD(sortColIdx, from->numCols *
sizeof(AttrNumber));
```

Dangerous Functions\Path 36:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=319
Status	New

The dangerous function, memcpy, was found in use at line 1007 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1018	1018
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copySort(const Sort *from)

```
....
1018.       COPY_POINTER_FIELD(sortOperators, from->numCols *
sizeof(Oid));
```

Dangerous Functions\Path 37:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=320
Status	New

The dangerous function, memcpy, was found in use at line 1007 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1019	1019
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copySort(const Sort *from)

```
....
1019.       COPY_POINTER_FIELD(collations, from->numCols * sizeof(Oid));
```

Dangerous Functions\Path 38:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=321
Status	New

The dangerous function, memcpy, was found in use at line 1007 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1020	1020
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copySort(const Sort *from)

```
....
1020.       COPY_POINTER_FIELD(nullsFirst, from->numCols *
sizeof(bool));
```

Dangerous Functions\Path 39:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=322
Status	New

The dangerous function, memcpy, was found in use at line 1030 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1037	1037
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyGroup(const Group *from)

```
....
1037.       COPY_POINTER_FIELD(grpColIdx, from->numCols *
sizeof(AttrNumber));
```

Dangerous Functions\Path 40:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=323
Status	New

The dangerous function, memcpy, was found in use at line 1030 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1038	1038
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyGroup(const Group *from)

```
....
1038.       COPY_POINTER_FIELD(grpOperators, from->numCols *
sizeof(Oid));
```

Dangerous Functions\Path 41:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=324
Status	New

The dangerous function, memcpy, was found in use at line 1047 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1058	1058
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyAgg(const Agg *from)

```
....
1058.             COPY_POINTER_FIELD(grpColIdx, from->numCols *
sizeof(AttrNumber));
```

Dangerous Functions\Path 42:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=325
Status	New

The dangerous function, memcpy, was found in use at line 1047 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1059	1059
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyAgg(const Agg *from)

```
....
1059.             COPY_POINTER_FIELD(grpOperators, from->numCols *
sizeof(Oid));
```

Dangerous Functions\Path 43:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=326
Status	New

The dangerous function, memcpy, was found in use at line 1076 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1086	1086
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyWindowAgg(const WindowAgg *from)

```
....
1086.             COPY_POINTER_FIELD(partColIdx, from->partNumCols *
sizeof(AttrNumber));
```

Dangerous Functions\Path 44:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=327
Status	New

The dangerous function, memcpy, was found in use at line 1076 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1087	1087
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyWindowAgg(const WindowAgg *from)

```
....
1087.             COPY_POINTER_FIELD(partOperators, from->partNumCols *
sizeof(Oid));
```

Dangerous Functions\Path 45:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=328
Status	New

The dangerous function, memcpy, was found in use at line 1076 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1092	1092
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyWindowAgg(const WindowAgg *from)

```
....
1092.               COPY_POINTER_FIELD(ordColIdx, from->ordNumCols *
sizeof(AttrNumber));
```

Dangerous Functions\Path 46:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=329
Status	New

The dangerous function, memcpy, was found in use at line 1076 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1093	1093
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyWindowAgg(const WindowAgg *from)

```
....
1093.               COPY_POINTER_FIELD(ordOperators, from->ordNumCols *
sizeof(Oid));
```

Dangerous Functions\Path 47:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=330
Status	New

The dangerous function, memcpy, was found in use at line 1111 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1124	1124
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyUnique(const Unique *from)

```
....
1124.       COPY_POINTER_FIELD(uniqColIdx, from->numCols *
sizeof(AttrNumber));
```

Dangerous Functions\Path 48:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=331
Status	New

The dangerous function, memcpy, was found in use at line 1111 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1125	1125
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyUnique(const Unique *from)

```
....
1125.       COPY_POINTER_FIELD(uniqOperators, from->numCols *
sizeof(Oid));
```

Dangerous Functions\Path 49:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=332
Status	New

The dangerous function, memcpy, was found in use at line 1158 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1173	1173
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copySetOp(const SetOp *from)

```
....
1173.       COPY_POINTER_FIELD(dupColIdx, from->numCols *
sizeof(AttrNumber));
```

Dangerous Functions\Path 50:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=333
Status	New

The dangerous function, memcpy, was found in use at line 1158 in PolarDB-for-PostgreSQL-2/copyfuncs.c file. Such functions may expose information and allow an attacker to get full control over the host machine.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1174	1174
Object	memcpy	memcpy

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copySetOp(const SetOp *from)

```
....
1174.       COPY_POINTER_FIELD(dupOperators, from->numCols *
sizeof(Oid));
```

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=1020028&projectid=20023&pathid=57
Status	New

The size of the buffer used by rebuild_database_list in avl_dbase, at line 952 of PolarDB-for-PostgreSQL-2/autovacuum.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that rebuild_database_list passes to avl_dbase, at line 952 of PolarDB-for-PostgreSQL-2/autovacuum.c, to overwrite the target buffer.

	Source	Destination

File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	1088	1088
Object	avl_dbase	avl_dbase

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method rebuild_database_list(Oid newdb)

```
....
1088.                   memcpy(&(dbary[i++]), db, sizeof(avl_dbase));
```

Buffer Overflow boundcpy WrongSizeParam\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=58
Status	New

The size of the buffer used by do_autovacuum in AutoVacOpts, at line 1968 of PolarDB-for-PostgreSQL-2/autovacuum.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that do_autovacuum passes to AutoVacOpts, at line 1968 of PolarDB-for-PostgreSQL-2/autovacuum.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	2168	2168
Object	AutoVacOpts	AutoVacOpts

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method do_autovacuum(void)

```
....
2168.                                           sizeof(AutoVacOpts));
```

Buffer Overflow boundcpy WrongSizeParam\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=59
Status	New

The size of the buffer used by extract_autovac_opts in AutoVacOpts, at line 2757 of PolarDB-for-PostgreSQL-2/autovacuum.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that extract_autovac_opts passes to AutoVacOpts, at line 2757 of PolarDB-for-PostgreSQL-2/autovacuum.c, to overwrite the target buffer.

	Source	Destination

File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	2771	2771
Object	AutoVacOpts	AutoVacOpts

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method extract_autovac_opts(HeapTuple tup, TupleDesc pg_class_desc)

```
....
2771.       memcpy(av, &(((StdRdOptions *) relopts)->autovacuum),
sizeof(AutoVacOpts));
```

Buffer Overflow boundcpy WrongSizeParam\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=60
Status	New

The size of the buffer used by _copyMotion in PlanSlice, at line 1409 of PolarDB-for-PostgreSQL-2/copyfuncs.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that _copyMotion passes to PlanSlice, at line 1409 of PolarDB-for-PostgreSQL-2/copyfuncs.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	1437	1437
Object	PlanSlice	PlanSlice

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyMotion(const Motion *from)

```
....
1437.             memcpy(newnode->senderSliceInfo, from-
>senderSliceInfo, sizeof(PlanSlice));
```

Buffer Overflow boundcpy WrongSizeParam\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=61
Status	New

The size of the buffer used by _copyForeignKeyCacheInfo in ->, at line 5045 of PolarDB-for-PostgreSQL-2/copyfuncs.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that _copyForeignKeyCacheInfo passes to ->, at line 5045 of PolarDB-for-PostgreSQL-2/copyfuncs.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	5054	5054
Object	->	->

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyForeignKeyCacheInfo(const ForeignKeyCacheInfo *from)

```
....
5054.       memcpy(newnode->conkey, from->conkey, sizeof(newnode-
>conkey));
```

Buffer Overflow boundcpy WrongSizeParam\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=62
Status	New

The size of the buffer used by _copyForeignKeyCacheInfo in ->, at line 5045 of PolarDB-for-PostgreSQL-2/copyfuncs.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that _copyForeignKeyCacheInfo passes to ->, at line 5045 of PolarDB-for-PostgreSQL-2/copyfuncs.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	5055	5055
Object	->	->

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyForeignKeyCacheInfo(const ForeignKeyCacheInfo *from)

```
....
5055.       memcpy(newnode->confkey, from->confkey, sizeof(newnode-
>confkey));
```

Buffer Overflow boundcpy WrongSizeParam\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=63
Status	New

The size of the buffer used by _copyForeignKeyCacheInfo in ->, at line 5045 of PolarDB-for-PostgreSQL-2/copyfuncs.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow

attack, using the source buffer that _copyForeignKeyCacheInfo passes to ->, at line 5045 of PolarDB-for-PostgreSQL-2/copyfuncs.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	5056	5056
Object	->	->

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyForeignKeyCacheInfo(const ForeignKeyCacheInfo *from)

```
....
5056.       memcpy(newnode->conpfeqop, from->conpfeqop, sizeof(newnode->conpfeqop));
```

Buffer Overflow boundcpy WrongSizeParam\Path 8:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=64
Status New

The size of the buffer used by EventTriggerCollectGrant in InternalGrant, at line 1838 of PolarDB-for-PostgreSQL-2/event_trigger.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that EventTriggerCollectGrant passes to InternalGrant, at line 1838 of PolarDB-for-PostgreSQL-2/event_trigger.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/event_trigger.c	PolarDB-for-PostgreSQL-2/event_trigger.c
Line	1856	1856
Object	InternalGrant	InternalGrant

Code Snippet
File Name PolarDB-for-PostgreSQL-2/event_trigger.c
Method EventTriggerCollectGrant(InternalGrant *istmt)

```
....
1856.       memcpy(icopy, istmt, sizeof(InternalGrant));
```

Buffer Overflow boundcpy WrongSizeParam\Path 9:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=65
Status New

The size of the buffer used by ExprEvalPushStep in ExprEvalStep, at line 2162 of PolarDB-for-PostgreSQL-2/execExpr.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ExprEvalPushStep passes to ExprEvalStep, at line 2162 of PolarDB-for-PostgreSQL-2/execExpr.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c
Line	2176	2176
Object	ExprEvalStep	ExprEvalStep

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExprEvalPushStep(ExprState *es, const ExprEvalStep *s)

```
....
2176.       memcpy(&es->steps[es->steps_len++], s,
sizeof(ExprEvalStep));
```

Buffer Overflow boundcpy WrongSizeParam\Path 10:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=66
Status New

The size of the buffer used by box_copy in BOX, at line 485 of PolarDB-for-PostgreSQL-2/geo_ops.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that box_copy passes to BOX, at line 485 of PolarDB-for-PostgreSQL-2/geo_ops.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/geo_ops.c	PolarDB-for-PostgreSQL-2/geo_ops.c
Line	489	489
Object	BOX	BOX

Code Snippet
File Name PolarDB-for-PostgreSQL-2/geo_ops.c
Method box_copy(BOX *box)

```
....
489.        memcpy((char *) result, (char *) box, sizeof(BOX));
```

Buffer Overflow boundcpy WrongSizeParam\Path 11:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=67
Status New

The size of the buffer used by close_lseg in Point, at line 2879 of PolarDB-for-PostgreSQL-2/geo_ops.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that close_lseg passes to Point, at line 2879 of PolarDB-for-PostgreSQL-2/geo_ops.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/geo_ops.c	PolarDB-for-PostgreSQL-2/geo_ops.c
Line	2890	2890
Object	Point	Point

Code Snippet
File Name PolarDB-for-PostgreSQL-2/geo_ops.c
Method close_lseg(PG_FUNCTION_ARGS)

```
....
2890.       memcpy(&point, &l1->p[0], sizeof(Point));
```

Buffer Overflow boundcpy WrongSizeParam\Path 12:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=68
Status New

The size of the buffer used by close_lseg in Point, at line 2879 of PolarDB-for-PostgreSQL-2/geo_ops.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that close_lseg passes to Point, at line 2879 of PolarDB-for-PostgreSQL-2/geo_ops.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/geo_ops.c	PolarDB-for-PostgreSQL-2/geo_ops.c
Line	2895	2895
Object	Point	Point

Code Snippet
File Name PolarDB-for-PostgreSQL-2/geo_ops.c
Method close_lseg(PG_FUNCTION_ARGS)

```
....
2895.             memcpy(&point, &l1->p[1], sizeof(Point));
```

Buffer Overflow boundcpy WrongSizeParam\Path 13:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=69
Status New

The size of the buffer used by close_lseg in Point, at line 2879 of PolarDB-for-PostgreSQL-2/geo_ops.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that close_lseg passes to Point, at line 2879 of PolarDB-for-PostgreSQL-2/geo_ops.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/geo_ops.c	PolarDB-for-PostgreSQL-2/geo_ops.c
Line	2903	2903
Object	Point	Point

Code Snippet
File Name PolarDB-for-PostgreSQL-2/geo_ops.c
Method close_lseg(PG_FUNCTION_ARGS)

```
....
2903.             memcpy(&point, result, sizeof(Point));
```

Buffer Overflow boundcpy WrongSizeParam\Path 14:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=70
Status New

The size of the buffer used by close_lseg in Point, at line 2879 of PolarDB-for-PostgreSQL-2/geo_ops.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that close_lseg passes to Point, at line 2879 of PolarDB-for-PostgreSQL-2/geo_ops.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/geo_ops.c	PolarDB-for-PostgreSQL-2/geo_ops.c
Line	2914	2914
Object	Point	Point

Code Snippet
File Name PolarDB-for-PostgreSQL-2/geo_ops.c
Method close_lseg(PG_FUNCTION_ARGS)

```
....
2914.             memcpy(&point, result, sizeof(Point));
```

Buffer Overflow boundcpy WrongSizeParam\Path 15:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=71
Status New

The size of the buffer used by circle_copy in CIRCLE, at line 4872 of PolarDB-for-PostgreSQL-2/geo_ops.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that circle_copy passes to CIRCLE, at line 4872 of PolarDB-for-PostgreSQL-2/geo_ops.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/geo_ops.c	PolarDB-for-PostgreSQL-2/geo_ops.c
Line	4880	4880
Object	CIRCLE	CIRCLE

Code Snippet
File Name PolarDB-for-PostgreSQL-2/geo_ops.c
Method circle_copy(CIRCLE *circle)

```
....
4880.       memcpy((char *) result, (char *) circle, sizeof(CIRCLE));
```

Buffer Overflow boundcpy WrongSizeParam\Path 16:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=72
Status New

The size of the buffer used by SerializeGUCState in actual_size, at line 13587 of PolarDB-for-PostgreSQL-2/guc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that SerializeGUCState passes to actual_size, at line 13587 of PolarDB-for-PostgreSQL-2/guc.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13604	13604
Object	actual_size	actual_size

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method SerializeGUCState(Size maxsize, char *start_address)

```
....
13604.      memcpy(start_address, &actual_size, sizeof(actual_size));
```

Buffer Overflow boundcpy WrongSizeParam\Path 17:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=73
Status New

The size of the buffer used by AddNewAttributeTuples in FormData_pg_attribute, at line 715 of PolarDB-for-PostgreSQL-2/heap.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that AddNewAttributeTuples passes to FormData_pg_attribute, at line 715 of PolarDB-for-PostgreSQL-2/heap.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/heap.c	PolarDB-for-PostgreSQL-2/heap.c
Line	792	792
Object	FormData_pg_attribute	FormData_pg_attribute

Code Snippet
File Name PolarDB-for-PostgreSQL-2/heap.c
Method AddNewAttributeTuples(Oid new_rel_oid,

```
....
792.                    memcpy(&attStruct, (char *) SysAtt[i],
sizeof(FormData_pg_attribute));
```

Buffer Overflow boundcpy WrongSizeParam\Path 18:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=74
Status	New

The size of the buffer used by LaunchParallelWorkers in int, at line 493 of PolarDB-for-PostgreSQL-2/parallel.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that LaunchParallelWorkers passes to int, at line 493 of PolarDB-for-PostgreSQL-2/parallel.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/parallel.c	PolarDB-for-PostgreSQL-2/parallel.c
Line	538	538
Object	int	int

Code Snippet
File Name PolarDB-for-PostgreSQL-2/parallel.c
Method LaunchParallelWorkers(ParallelContext *pcxt)

```
....
538.              memcpy(worker.bgw_extra, &i, sizeof(int));
```

Buffer Overflow boundcpy WrongSizeParam\Path 19:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=75
Status	New

The size of the buffer used by ParallelWorkerMain in int, at line 1209 of PolarDB-for-PostgreSQL-2/parallel.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ParallelWorkerMain passes to int, at line 1209 of PolarDB-for-PostgreSQL-2/parallel.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/parallel.c	PolarDB-for-PostgreSQL-2/parallel.c
Line	1240	1240
Object	int	int

Code Snippet
File Name PolarDB-for-PostgreSQL-2/parallel.c
Method ParallelWorkerMain(Datum main_arg)

```
....
1240.       memcpy(&ParallelWorkerNumber, MyBgworkerEntry->bgw_extra,
sizeof(int));
```

Buffer Overflow boundcpy WrongSizeParam\Path 20:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=76
Status New

The size of the buffer used by polar_hold_shared_storage in uint32_t, at line 95 of PolarDB-for-PostgreSQL-2/polar_io_fencing.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that polar_hold_shared_storage passes to uint32_t, at line 95 of PolarDB-for-PostgreSQL-2/polar_io_fencing.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/polar_io_fencing.c	PolarDB-for-PostgreSQL-2/polar_io_fencing.c
Line	108	108
Object	uint32_t	uint32_t

Code Snippet
File Name PolarDB-for-PostgreSQL-2/polar_io_fencing.c
Method polar_hold_shared_storage(bool force_hold)

```
....
108.              memcpy(polar_rwid.random_id + i, &seed,
sizeof(uint32_t));
```

Buffer Overflow boundcpy WrongSizeParam\Path 21:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=77

Status	New

The size of the buffer used by ShowUsage in user, at line 4973 of PolarDB-for-PostgreSQL-2/postgres.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ShowUsage passes to user, at line 4973 of PolarDB-for-PostgreSQL-2/postgres.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	4983	4983
Object	user	user

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres.c
Method ShowUsage(const char *title)

```
....
4983.       memcpy((char *) &user, (char *) &r.ru_utime, sizeof(user));
```

Buffer Overflow boundcpy WrongSizeParam\Path 22:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=78
Status	New

The size of the buffer used by ShowUsage in sys, at line 4973 of PolarDB-for-PostgreSQL-2/postgres.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ShowUsage passes to sys, at line 4973 of PolarDB-for-PostgreSQL-2/postgres.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	4984	4984
Object	sys	sys

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres.c
Method ShowUsage(const char *title)

```
....
4984.       memcpy((char *) &sys, (char *) &r.ru_stime, sizeof(sys));
```

Buffer Overflow boundcpy WrongSizeParam\Path 23:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=79
Status	New

The size of the buffer used by PxDispatchPlan in PlannedStmt, at line 149 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that PxDispatchPlan passes to PlannedStmt, at line 149 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	172	172
Object	PlannedStmt	PlannedStmt

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method PxDispatchPlan(struct QueryDesc *queryDesc,

```
....
172.        memcpy(stmt, queryDesc->plannedstmt, sizeof(PlannedStmt));
```

Buffer Overflow boundcpy WrongSizeParam\Path 24:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=80
Status New

The size of the buffer used by buildPXQueryString in len, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that buildPXQueryString passes to len, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	528	528
Object	len	len

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method buildPXQueryString(DispatchCommandQueryParms *pQueryParms, int *finalLen)

```
....
528.        INT32_ENCODE(len);
```

Buffer Overflow boundcpy WrongSizeParam\Path 25:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=81

Status	New

The size of the buffer used by buildPXQueryString in px_serialize_version, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that buildPXQueryString passes to px_serialize_version, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	529	529
Object	px_serialize_version	px_serialize_version

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method buildPXQueryString(DispatchCommandQueryParms *pQueryParms, int *finalLen)

```
....
529.        INT32_ENCODE(px_serialize_version);
```

Buffer Overflow boundcpy WrongSizeParam\Path 26:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=82
Status	New

The size of the buffer used by buildPXQueryString in sessionUserId, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that buildPXQueryString passes to sessionUserId, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	530	530
Object	sessionUserId	sessionUserId

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method buildPXQueryString(DispatchCommandQueryParms *pQueryParms, int *finalLen)

```
....
530.        INT32_ENCODE(sessionUserId);
```

Buffer Overflow boundcpy WrongSizeParam\Path 27:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=200

	23&pathid=83
Status	New

The size of the buffer used by buildPXQueryString in outerUserId, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that buildPXQueryString passes to outerUserId, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	531	531
Object	outerUserId	outerUserId

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method buildPXQueryString(DispatchCommandQueryParms *pQueryParms, int *finalLen)

```
....
531.        INT32_ENCODE(outerUserId);
```

Buffer Overflow boundcpy WrongSizeParam\Path 28:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=84
Status	New

The size of the buffer used by buildPXQueryString in currentUserId, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that buildPXQueryString passes to currentUserId, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	532	532
Object	currentUserId	currentUserId

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method buildPXQueryString(DispatchCommandQueryParms *pQueryParms, int *finalLen)

```
....
532.        INT32_ENCODE(currentUserId);
```

Buffer Overflow boundcpy WrongSizeParam\Path 29:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=85
Status	New

The size of the buffer used by buildPXQueryString in u32, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that buildPXQueryString passes to u32, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	534	534
Object	u32	u32

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method buildPXQueryString(DispatchCommandQueryParms *pQueryParms, int *finalLen)

```
....
534.        UINT64_ENCODE(currentStatementStartTimestamp);
```

Buffer Overflow boundcpy WrongSizeParam\Path 30:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=86
Status	New

The size of the buffer used by buildPXQueryString in u32, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that buildPXQueryString passes to u32, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	534	534
Object	u32	u32

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method buildPXQueryString(DispatchCommandQueryParms *pQueryParms, int *finalLen)

```
....
534.        UINT64_ENCODE(currentStatementStartTimestamp);
```

Buffer Overflow boundcpy WrongSizeParam\Path 31:

Severity	Medium
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=87
Status	New

The size of the buffer used by buildPXQueryString in u32, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that buildPXQueryString passes to u32, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	535	535
Object	u32	u32

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method buildPXQueryString(DispatchCommandQueryParms *pQueryParms, int *finalLen)

```
....
535.        UINT64_ENCODE(sql_trace_id.uval);
```

Buffer Overflow boundcpy WrongSizeParam\Path 32:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=88
Status	New

The size of the buffer used by buildPXQueryString in u32, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that buildPXQueryString passes to u32, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	535	535
Object	u32	u32

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method buildPXQueryString(DispatchCommandQueryParms *pQueryParms, int *finalLen)

```
....
535.        UINT64_ENCODE(sql_trace_id.uval);
```

Buffer Overflow boundcpy WrongSizeParam\Path 33:

Severity	Medium

Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=89
Status	New

The size of the buffer used by buildPXQueryString in command_len, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that buildPXQueryString passes to command_len, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	537	537
Object	command_len	command_len

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method buildPXQueryString(DispatchCommandQueryParms *pQueryParms, int *finalLen)

```
....
537.        STRN_ENCODE(command_len, command);
```

Buffer Overflow boundcpy WrongSizeParam\Path 34:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=90
Status	New

The size of the buffer used by buildPXQueryString in querytree_len, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that buildPXQueryString passes to querytree_len, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	538	538
Object	querytree_len	querytree_len

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method buildPXQueryString(DispatchCommandQueryParms *pQueryParms, int *finalLen)

```
....
538.        STR_ENCODE(querytree_len, querytree);
```

Buffer Overflow boundcpy WrongSizeParam\Path 35:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=91
Status	New

The size of the buffer used by buildPXQueryString in plantree_len, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that buildPXQueryString passes to plantree_len, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	539	539
Object	plantree_len	plantree_len

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method buildPXQueryString(DispatchCommandQueryParms *pQueryParms, int *finalLen)

```
....
539.        STR_ENCODE(plantree_len, plantree);
```

Buffer Overflow boundcpy WrongSizeParam\Path 36:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=92
Status	New

The size of the buffer used by buildPXQueryString in params_len, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that buildPXQueryString passes to params_len, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	540	540
Object	params_len	params_len

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method buildPXQueryString(DispatchCommandQueryParms *pQueryParms, int *finalLen)

```
....
540.        STR_ENCODE(params_len, params);
```

Buffer Overflow boundcpy WrongSizeParam\Path 37:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=93
Status	New

The size of the buffer used by buildPXQueryString in sddesc_len, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that buildPXQueryString passes to sddesc_len, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	541	541
Object	sddesc_len	sddesc_len

Code Snippet

File Name PolarDB-for-PostgreSQL-2/px_disp_query.c

Method buildPXQueryString(DispatchCommandQueryParms *pQueryParms, int *finalLen)

```
....
541.        STR_ENCODE(sddesc_len, sddesc);
```

Buffer Overflow boundcpy WrongSizeParam\Path 38:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=94
Status	New

The size of the buffer used by buildPXQueryString in sdsnapshot_len, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that buildPXQueryString passes to sdsnapshot_len, at line 425 of PolarDB-for-PostgreSQL-2/px_disp_query.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	542	542
Object	sdsnapshot_len	sdsnapshot_len

Code Snippet

File Name PolarDB-for-PostgreSQL-2/px_disp_query.c

Method buildPXQueryString(DispatchCommandQueryParms *pQueryParms, int *finalLen)

```
....
542.        STR_ENCODE(sdsnapshot_len, sdsnapshot);
```

Buffer Overflow boundcpy WrongSizeParam\Path 39:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=95
Status	New

The size of the buffer used by RI_Initial_Check in RI_ConstraintInfo, at line 1849 of PolarDB-for-PostgreSQL-2/ri_triggers.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that RI_Initial_Check passes to RI_ConstraintInfo, at line 1849 of PolarDB-for-PostgreSQL-2/ri_triggers.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ri_triggers.c	PolarDB-for-PostgreSQL-2/ri_triggers.c
Line	2072	2072
Object	RI_ConstraintInfo	RI_ConstraintInfo

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ri_triggers.c
Method RI_Initial_Check(Trigger *trigger, Relation fk_rel, Relation pk_rel)

```
....
2072.             memcpy(&fake_riinfo, riinfo,
sizeof(RI_ConstraintInfo));
```

Buffer Overflow boundcpy WrongSizeParam\Path 40:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=96
Status	New

The size of the buffer used by ri_LoadConstraintInfo in NameData, at line 2338 of PolarDB-for-PostgreSQL-2/ri_triggers.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that ri_LoadConstraintInfo passes to NameData, at line 2338 of PolarDB-for-PostgreSQL-2/ri_triggers.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ri_triggers.c	PolarDB-for-PostgreSQL-2/ri_triggers.c
Line	2378	2378
Object	NameData	NameData

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ri_triggers.c
Method ri_LoadConstraintInfo(Oid constraintOid)

```
....
2378.       memcpy(&riinfo->conname, &conForm->conname,
sizeof(NameData));
```

Buffer Overflow boundcpy WrongSizeParam\Path 41:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=97
Status	New

The size of the buffer used by CopyTriggerDesc in TriggerDesc, at line 2142 of PolarDB-for-PostgreSQL-2/trigger.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that CopyTriggerDesc passes to TriggerDesc, at line 2142 of PolarDB-for-PostgreSQL-2/trigger.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/trigger.c	PolarDB-for-PostgreSQL-2/trigger.c
Line	2152	2152
Object	TriggerDesc	TriggerDesc

Code Snippet
File Name PolarDB-for-PostgreSQL-2/trigger.c
Method CopyTriggerDesc(TriggerDesc *trigdesc)

```
....
2152.       memcpy(newdesc, trigdesc, sizeof(TriggerDesc));
```

Buffer Overflow boundcpy WrongSizeParam\Path 42:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=98
Status	New

The size of the buffer used by array_cat in ndims, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that array_cat passes to ndims, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	366	366
Object	ndims	ndims

Code Snippet
File Name PolarDB-for-PostgreSQL-2/array_userfuncs.c

Method	array_cat(PG_FUNCTION_ARGS)

```
....
366.                memcpy(dims, dims2, ndims * sizeof(int));
```

Buffer Overflow boundcpy WrongSizeParam\Path 43:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=99
Status	New

The size of the buffer used by array_cat in int, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that array_cat passes to int, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	366	366
Object	int	int

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method	array_cat(PG_FUNCTION_ARGS)

```
....
366.                memcpy(dims, dims2, ndims * sizeof(int));
```

Buffer Overflow boundcpy WrongSizeParam\Path 44:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=100
Status	New

The size of the buffer used by array_cat in ndims, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that array_cat passes to ndims, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	367	367
Object	ndims	ndims

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method	array_cat(PG_FUNCTION_ARGS)

```
....
367.                memcpy(lbs, lbs2, ndims * sizeof(int));
```

Buffer Overflow boundcpy WrongSizeParam\Path 45:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=101
Status	New

The size of the buffer used by array_cat in int, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that array_cat passes to int, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	367	367
Object	int	int

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method	array_cat(PG_FUNCTION_ARGS)

```
....
367.                memcpy(lbs, lbs2, ndims * sizeof(int));
```

Buffer Overflow boundcpy WrongSizeParam\Path 46:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=102
Status	New

The size of the buffer used by array_cat in ndims, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that array_cat passes to ndims, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	394	394
Object	ndims	ndims

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method	array_cat(PG_FUNCTION_ARGS)

```
....
394.                memcpy(dims, dims1, ndims * sizeof(int));
```

Buffer Overflow boundcpy WrongSizeParam\Path 47:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=103
Status	New

The size of the buffer used by array_cat in int, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that array_cat passes to int, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	394	394
Object	int	int

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method	array_cat(PG_FUNCTION_ARGS)

```
....
394.                memcpy(dims, dims1, ndims * sizeof(int));
```

Buffer Overflow boundcpy WrongSizeParam\Path 48:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=104
Status	New

The size of the buffer used by array_cat in ndims, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that array_cat passes to ndims, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	395	395
Object	ndims	ndims

Code Snippet
File Name PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method array_cat(PG_FUNCTION_ARGS)

```
....
395.                memcpy(lbs, lbs1, ndims * sizeof(int));
```

Buffer Overflow boundcpy WrongSizeParam\Path 49:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=105
Status	New

The size of the buffer used by array_cat in int, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that array_cat passes to int, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	395	395
Object	int	int

Code Snippet
File Name PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method array_cat(PG_FUNCTION_ARGS)

```
....
395.                memcpy(lbs, lbs1, ndims * sizeof(int));
```

Buffer Overflow boundcpy WrongSizeParam\Path 50:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=106
Status	New

The size of the buffer used by array_cat in ndims, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that array_cat passes to ndims, at line 218 of PolarDB-for-PostgreSQL-2/array_userfuncs.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	433	433

Object	ndims	ndims

Code Snippet
File Name PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method array_cat(PG_FUNCTION_ARGS)

```
....
433.        memcpy(ARR_DIMS(result), dims, ndims * sizeof(int));
```

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=1020028&projectid=20023&pathid=662
Status	New

The variable declared in dtablespacename at PolarDB-for-PostgreSQL-2/dbcommands.c in line 109 is not initialized when it is used by polar_dstpath at PolarDB-for-PostgreSQL-2/dbcommands.c in line 109.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/dbcommands.c	PolarDB-for-PostgreSQL-2/dbcommands.c
Line	132	661
Object	dtablespacename	polar_dstpath

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbcommands.c
Method createdb(ParseState *pstate, const CreatedbStmt *stmt)

```
....
132.        DefElem    *dtablespacename = NULL;
....
661.                            polar_dstpath =
polar_get_database_path(dboid, dsttablespace);
```

Use of Zero Initialized Pointer\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=663
Status	New

The variable declared in downer at PolarDB-for-PostgreSQL-2/dbcommands.c in line 109 is not initialized when it is used by dbowner at PolarDB-for-PostgreSQL-2/dbcommands.c in line 109.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/dbcommands.c	PolarDB-for-PostgreSQL-2/dbcommands.c
Line	133	257
Object	downer	dbowner

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbcommands.c
Method createdb(ParseState *pstate, const CreatedbStmt *stmt)

```
....
133.        DefElem    *downer = NULL;
....
257.              dbowner = defGetString(downer);
```

Use of Zero Initialized Pointer\Path 3:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=664
Status New

The variable declared in dtemplate at PolarDB-for-PostgreSQL-2/dbcommands.c in line 109 is not initialized when it is used by dbtemplate at PolarDB-for-PostgreSQL-2/dbcommands.c in line 109.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/dbcommands.c	PolarDB-for-PostgreSQL-2/dbcommands.c
Line	134	259
Object	dtemplate	dbtemplate

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbcommands.c
Method createdb(ParseState *pstate, const CreatedbStmt *stmt)

```
....
134.        DefElem    *dtemplate = NULL;
....
259.              dbtemplate = defGetString(dtemplate);
```

Use of Zero Initialized Pointer\Path 4:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=200

	23&pathid=665
Status	New

The variable declared in dcollate at PolarDB-for-PostgreSQL-2/dbcommands.c in line 109 is not initialized when it is used by dbcollate at PolarDB-for-PostgreSQL-2/dbcommands.c in line 109.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/dbcommands.c	PolarDB-for-PostgreSQL-2/dbcommands.c
Line	136	289
Object	dcollate	dbcollate

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbcommands.c
Method createdb(ParseState *pstate, const CreatedbStmt *stmt)

```
....
136.        DefElem    *dcollate = NULL;
....
289.              dbcollate = defGetString(dcollate);
```

Use of Zero Initialized Pointer\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=666
Status	New

The variable declared in dctype at PolarDB-for-PostgreSQL-2/dbcommands.c in line 109 is not initialized when it is used by dbctype at PolarDB-for-PostgreSQL-2/dbcommands.c in line 109.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/dbcommands.c	PolarDB-for-PostgreSQL-2/dbcommands.c
Line	137	291
Object	dctype	dbctype

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbcommands.c
Method createdb(ParseState *pstate, const CreatedbStmt *stmt)

```
....
137.        DefElem    *dctype = NULL;
....
291.              dbctype = defGetString(dctype);
```

Use of Zero Initialized Pointer\Path 6:

Severity	Medium
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=667
Status	New

The variable declared in result at PolarDB-for-PostgreSQL-2/dml.c in line 93 is not initialized when it is used by result at PolarDB-for-PostgreSQL-2/dml.c in line 93.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/dml.c	PolarDB-for-PostgreSQL-2/dml.c
Line	95	116
Object	result	result

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dml.c
Method fixup_inherited_columns(Oid parentId, Oid childId, Bitmapset *columns)

```
....
95.   Bitmapset  *result = NULL;
....
116.                    result = bms_add_member(result, index);
```

Use of Zero Initialized Pointer\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=668
Status	New

The variable declared in subplan at PolarDB-for-PostgreSQL-2/execExpr.c in line 2394 is not initialized when it is used by junkFilter at PolarDB-for-PostgreSQL-2/execExpr.c in line 2394.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c
Line	2420	2454
Object	subplan	junkFilter

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExecInitWholeRowVar(ExprEvalStep *scratch, Var *variable, ExprState *state)

```
....
2420.            PlanState  *subplan = NULL;
....
2454.                        scratch->d.wholerow.junkFilter =
```

Use of Zero Initialized Pointer\Path 8:

Severity	Medium
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=669
Status	New

The variable declared in str at PolarDB-for-PostgreSQL-2/guc.c in line 13203 is not initialized when it is used by str at PolarDB-for-PostgreSQL-2/guc.c in line 13203.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13208	13222
Object	str	str

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method read_string_with_null(FILE *fp)

```
....
13208.      char     *str = NULL;
....
13222.                  str = guc_realloc(FATAL, str, maxlen *= 2);
```

Use of Zero Initialized Pointer\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=670
Status	New

The variable declared in lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194 is not initialized when it is used by lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ltree_io.c	PolarDB-for-PostgreSQL-2/ltree_io.c
Line	203	268
Object	lptr	lptr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ltree_io.c
Method lquery_in(PG_FUNCTION_ARGS)

```
....
203.        nodeitem   *lptr = NULL;
....
268.                        lptr++;
```

Use of Zero Initialized Pointer\Path 10:

Severity	Medium
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=671
Status	New

The variable declared in lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194 is not initialized when it is used by lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ltree_io.c	PolarDB-for-PostgreSQL-2/ltree_io.c
Line	203	282
Object	lptr	lptr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ltree_io.c
Method lquery_in(PG_FUNCTION_ARGS)

```
....
203.        nodeitem   *lptr = NULL;
....
282.                           lptr->flag |= LVAR_INCASE;
```

Use of Zero Initialized Pointer\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=672
Status	New

The variable declared in lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194 is not initialized when it is used by lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ltree_io.c	PolarDB-for-PostgreSQL-2/ltree_io.c
Line	203	289
Object	lptr	lptr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ltree_io.c
Method lquery_in(PG_FUNCTION_ARGS)

```
....
203.        nodeitem   *lptr = NULL;
....
289.                           lptr->flag |= LVAR_ANYEND;
```

Use of Zero Initialized Pointer\Path 12:

Severity	Medium
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=673
Status	New

The variable declared in lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194 is not initialized when it is used by lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ltree_io.c	PolarDB-for-PostgreSQL-2/ltree_io.c
Line	203	296
Object	lptr	lptr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ltree_io.c
Method lquery_in(PG_FUNCTION_ARGS)

```
....
203.        nodeitem   *lptr = NULL;
....
296.                           lptr->flag |= LVAR_SUBLEXEME;
```

Use of Zero Initialized Pointer\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=674
Status	New

The variable declared in lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194 is not initialized when it is used by lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ltree_io.c	PolarDB-for-PostgreSQL-2/ltree_io.c
Line	203	334
Object	lptr	lptr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ltree_io.c
Method lquery_in(PG_FUNCTION_ARGS)

```
....
203.        nodeitem   *lptr = NULL;
....
334.                           if (lptr->flag)
```

Use of Zero Initialized Pointer\Path 14:

Severity	Medium
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=675
Status	New

The variable declared in lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194 is not initialized when it is used by lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ltree_io.c	PolarDB-for-PostgreSQL-2/ltree_io.c
Line	203	301
Object	lptr	lptr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ltree_io.c
Method lquery_in(PG_FUNCTION_ARGS)

```
....
203.        nodeitem   *lptr = NULL;
....
301.                           lptr->len = ptr - lptr->start -
```

Use of Zero Initialized Pointer\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=676
Status	New

The variable declared in lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194 is not initialized when it is used by lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ltree_io.c	PolarDB-for-PostgreSQL-2/ltree_io.c
Line	203	317
Object	lptr	lptr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ltree_io.c
Method lquery_in(PG_FUNCTION_ARGS)

```
....
203.        nodeitem   *lptr = NULL;
....
317.                           lptr->len = ptr - lptr->start -
```

Use of Zero Initialized Pointer\Path 16:

Severity	Medium
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=677
Status	New

The variable declared in lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194 is not initialized when it is used by lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ltree_io.c	PolarDB-for-PostgreSQL-2/ltree_io.c
Line	203	446
Object	lptr	lptr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ltree_io.c
Method lquery_in(PG_FUNCTION_ARGS)

```
....
203.        nodeitem   *lptr = NULL;
....
446.              lptr->len = ptr - lptr->start -
```

Use of Zero Initialized Pointer\Path 17:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=678
Status	New

The variable declared in astate at PolarDB-for-PostgreSQL-2/misc.c in line 806 is not initialized when it is used by astate at PolarDB-for-PostgreSQL-2/misc.c in line 806.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/misc.c	PolarDB-for-PostgreSQL-2/misc.c
Line	811	860
Object	astate	astate

Code Snippet
File Name PolarDB-for-PostgreSQL-2/misc.c
Method parse_ident(PG_FUNCTION_ARGS)

```
....
811.        ArrayBuildState *astate = NULL;
....
860.                    astate = accumArrayResult(astate,
CStringGetTextDatum(curname),
```

Use of Zero Initialized Pointer\Path 18:

Severity	Medium

Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=679
Status	New

The variable declared in astate at PolarDB-for-PostgreSQL-2/misc.c in line 806 is not initialized when it is used by astate at PolarDB-for-PostgreSQL-2/misc.c in line 806.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/misc.c	PolarDB-for-PostgreSQL-2/misc.c
Line	811	884
Object	astate	astate

Code Snippet
File Name PolarDB-for-PostgreSQL-2/misc.c
Method parse_ident(PG_FUNCTION_ARGS)

```
....
811.        ArrayBuildState *astate = NULL;
....
884.                    astate = accumArrayResult(astate,
PointerGetDatum(part), false,
```

Use of Zero Initialized Pointer\Path 19:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=680
Status	New

The variable declared in astate at PolarDB-for-PostgreSQL-2/nodeSubplan.c in line 225 is not initialized when it is used by astate at PolarDB-for-PostgreSQL-2/nodeSubplan.c in line 225.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeSubplan.c	PolarDB-for-PostgreSQL-2/nodeSubplan.c
Line	238	381
Object	astate	astate

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeSubplan.c
Method ExecScanSubPlan(SubPlanState *node,

```
....
238.        ArrayBuildStateAny *astate = NULL;
....
381.                    astate = accumArrayResultAny(astate, dvalue,
disnull,
```

Use of Zero Initialized Pointer\Path 20:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=681
Status	New

The variable declared in astate at PolarDB-for-PostgreSQL-2/nodeSubplan.c in line 1046 is not initialized when it is used by astate at PolarDB-for-PostgreSQL-2/nodeSubplan.c in line 1046.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeSubplan.c	PolarDB-for-PostgreSQL-2/nodeSubplan.c
Line	1058	1140
Object	astate	astate

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeSubplan.c
Method ExecSetParamPlan(SubPlanState *node, ExprContext *econtext)

```
....
1058.       ArrayBuildStateAny *astate = NULL;
....
1140.                   astate = accumArrayResultAny(astate, dvalue,
disnull,
```

Use of Zero Initialized Pointer\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=682
Status	New

The variable declared in replication_state at PolarDB-for-PostgreSQL-2/origin.c in line 858 is not initialized when it is used by replication_state at PolarDB-for-PostgreSQL-2/origin.c in line 858.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/origin.c	PolarDB-for-PostgreSQL-2/origin.c
Line	863	939
Object	replication_state	replication_state

Code Snippet
File Name PolarDB-for-PostgreSQL-2/origin.c
Method replorigin_advance(RepOriginId node,

```
....
863.        ReplicationState *replication_state = NULL;
....
939.        Assert(replication_state->roident != InvalidRepOriginId);
```

Use of Zero Initialized Pointer\Path 22:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=683
Status	New

The variable declared in subref at PolarDB-for-PostgreSQL-2/plperl.c in line 2087 is not initialized when it is used by reference at PolarDB-for-PostgreSQL-2/plperl.c in line 2087.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/plperl.c	PolarDB-for-PostgreSQL-2/plperl.c
Line	2093	2151
Object	subref	reference

Code Snippet
File Name PolarDB-for-PostgreSQL-2/plperl.c
Method plperl_create_sub(plperl_proc_desc *prodesc, const char *s, Oid fn_oid)

```
....
2093.        SV             *subref = NULL;
....
2151.        prodesc->reference = subref;
```

Use of Zero Initialized Pointer\Path 23:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=684
Status	New

The variable declared in pformats at PolarDB-for-PostgreSQL-2/postgres.c in line 1665 is not initialized when it is used by pformats at PolarDB-for-PostgreSQL-2/postgres.c in line 1665.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	1670	1884
Object	pformats	pformats

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres.c
Method exec_bind_message(StringInfo input_message)

```
....
1670.       int16    *pformats = NULL;
....
1884.                           pformat = pformats[paramno];
```

Use of Zero Initialized Pointer\Path 24:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=685
Status	New

The variable declared in pformats at PolarDB-for-PostgreSQL-2/postgres.c in line 1665 is not initialized when it is used by pformats at PolarDB-for-PostgreSQL-2/postgres.c in line 1665.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	1670	1886
Object	pformats	pformats

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres.c
Method exec_bind_message(StringInfo input_message)

```
....
1670.       int16    *pformats = NULL;
....
1886.                           pformat = pformats[0];
```

Use of Zero Initialized Pointer\Path 25:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=686
Status	New

The variable declared in plan at PolarDB-for-PostgreSQL-2/postgres_px.c in line 42 is not initialized when it is used by plan at PolarDB-for-PostgreSQL-2/postgres_px.c in line 42.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres_px.c	PolarDB-for-PostgreSQL-2/postgres_px.c
Line	54	199
Object	plan	plan

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres_px.c
Method exec_px_query(const char *query_string,

```
....
54.   PlannedStmt    *plan = NULL;
....
199.          commandType = plan->commandType;
```

Use of Zero Initialized Pointer\Path 26:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=687
Status	New

The variable declared in slice at PolarDB-for-PostgreSQL-2/postgres_px.c in line 42 is not initialized when it is used by slice at PolarDB-for-PostgreSQL-2/postgres_px.c in line 42.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres_px.c	PolarDB-for-PostgreSQL-2/postgres_px.c
Line	58	160
Object	slice	slice

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres_px.c
Method exec_px_query(const char *query_string,

```
....
58.   ExecSlice  *slice = NULL;
....
160.                    sliceTable->localSlice = slice->sliceIndex;
```

Use of Zero Initialized Pointer\Path 27:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=688
Status	New

The variable declared in pQueryParms at PolarDB-for-PostgreSQL-2/px_disp_query.c in line 184 is not initialized when it is used by pQueryParms at PolarDB-for-PostgreSQL-2/px_disp_query.c in line 184.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	198	201
Object	pQueryParms	pQueryParms

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c

Method	pxdisp_buildPlanQueryParms(struct QueryDesc *queryDesc,

```
....
198.        DispatchCommandQueryParms *pQueryParms = NULL;
....
201.        pQueryParms = (DispatchCommandQueryParms *)
palloc0(sizeof(*pQueryParms));
```

Use of Zero Initialized Pointer\Path 28:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=689
Status	New

The variable declared in err at PolarDB-for-PostgreSQL-2/subscriptioncmds.c in line 808 is not initialized when it is used by wrconn at PolarDB-for-PostgreSQL-2/subscriptioncmds.c in line 808.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/subscriptioncmds.c	PolarDB-for-PostgreSQL-2/subscriptioncmds.c
Line	822	968
Object	err	wrconn

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/subscriptioncmds.c
Method	DropSubscription(DropSubscriptionStmt *stmt, bool isTopLevel)

```
....
822.        char     *err = NULL;
....
968.        wrconn = walrcv_connect(conninfo, true, subname, &err);
```

Use of Zero Initialized Pointer\Path 29:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=690
Status	New

The variable declared in rettuple at PolarDB-for-PostgreSQL-2/tsvector_op.c in line 2554 is not initialized when it is used by rettuple at PolarDB-for-PostgreSQL-2/tsvector_op.c in line 2554.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/tsvector_op.c	PolarDB-for-PostgreSQL-2/tsvector_op.c
Line	2559	2686
Object	rettuple	rettuple

Code Snippet
File Name PolarDB-for-PostgreSQL-2/tsvector_op.c
Method tsvector_update_trigger(PG_FUNCTION_ARGS, bool config_column)

```
....
2559.       HeapTuple   rettuple = NULL;
....
2686.       rettuple = heap_modify_tuple_by_cols(rettuple, rel->rd_att,
```

Use of Zero Initialized Pointer\Path 30:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=691
Status	New

The variable declared in allpos at PolarDB-for-PostgreSQL-2/tsvector_op.c in line 1324 is not initialized when it is used by allpos at PolarDB-for-PostgreSQL-2/tsvector_op.c in line 1324.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/tsvector_op.c	PolarDB-for-PostgreSQL-2/tsvector_op.c
Line	1358	1395
Object	allpos	allpos

Code Snippet
File Name PolarDB-for-PostgreSQL-2/tsvector_op.c
Method checkcondition_str(void *checkval, QueryOperand *val, ExecPhraseData *data)

```
....
1358.             WordEntryPos *allpos = NULL;
....
1395.                                             allpos =
repalloc(allpos, sizeof(WordEntryPos) * totalpos);
```

Use of Zero Initialized Pointer\Path 31:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=692
Status	New

The variable declared in allpos at PolarDB-for-PostgreSQL-2/tsvector_op.c in line 1324 is not initialized when it is used by allpos at PolarDB-for-PostgreSQL-2/tsvector_op.c in line 1324.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/tsvector_op.c	PolarDB-for-PostgreSQL-2/tsvector_op.c
Line	1358	1399
Object	allpos	allpos

Code Snippet
File Name PolarDB-for-PostgreSQL-2/tsvector_op.c
Method checkcondition_str(void *checkval, QueryOperand *val, ExecPhraseData *data)

```
....
1358.             WordEntryPos *allpos = NULL;
....
1399.                                 memcpy(allpos + npos, data->pos,
sizeof(WordEntryPos) * data->npos);
```

Use of Zero Initialized Pointer\Path 32:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=693
Status	New

The variable declared in != at PolarDB-for-PostgreSQL-2/xact.c in line 524 is not initialized when it is used by CurrentResourceOwner at PolarDB-for-PostgreSQL-2/xact.c in line 524.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/xact.c	PolarDB-for-PostgreSQL-2/xact.c
Line	526	626
Object	!=	CurrentResourceOwner

Code Snippet
File Name PolarDB-for-PostgreSQL-2/xact.c
Method AssignTransactionId(TransactionState s)

```
....
526.        bool        isSubXact = (s->parent != NULL);
....
626.        CurrentResourceOwner = s->curTransactionOwner;
```

Use of Zero Initialized Pointer\Path 33:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=694
Status	New

The variable declared in str at PolarDB-for-PostgreSQL-2/_int_bool.c in line 478 is not initialized when it is used by str at PolarDB-for-PostgreSQL-2/_int_bool.c in line 478.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/_int_bool.c	PolarDB-for-PostgreSQL-2/_int_bool.c
Line	497	524

Object	str	str

Code Snippet
File Name PolarDB-for-PostgreSQL-2/_int_bool.c
Method bqarr_in(PG_FUNCTION_ARGS)

```
....
497.        state.str = NULL;
....
524.              state.str = tmp;
```

Use of Zero Initialized Pointer\Path 34:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=695
Status	New

The variable declared in str at PolarDB-for-PostgreSQL-2/_int_bool.c in line 478 is not initialized when it is used by str at PolarDB-for-PostgreSQL-2/_int_bool.c in line 478.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/_int_bool.c	PolarDB-for-PostgreSQL-2/_int_bool.c
Line	497	522
Object	str	str

Code Snippet
File Name PolarDB-for-PostgreSQL-2/_int_bool.c
Method bqarr_in(PG_FUNCTION_ARGS)

```
....
497.        state.str = NULL;
....
522.              tmp = state.str->next;
```

Use of Zero Initialized Pointer\Path 35:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=696
Status	New

The variable declared in str at PolarDB-for-PostgreSQL-2/_int_bool.c in line 478 is not initialized when it is used by str at PolarDB-for-PostgreSQL-2/_int_bool.c in line 478.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/_int_bool.c	PolarDB-for-PostgreSQL-2/_int_bool.c

Line	497	521
Object	str	str

Code Snippet
File Name PolarDB-for-PostgreSQL-2/_int_bool.c
Method bqarr_in(PG_FUNCTION_ARGS)

```
....
497.        state.str = NULL;
....
521.              ptr[i].val = state.str->val;
```

Use of Zero Initialized Pointer\Path 36:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=697
Status New

The variable declared in str at PolarDB-for-PostgreSQL-2/_int_bool.c in line 478 is not initialized when it is used by str at PolarDB-for-PostgreSQL-2/_int_bool.c in line 478.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/_int_bool.c	PolarDB-for-PostgreSQL-2/_int_bool.c
Line	497	520
Object	str	str

Code Snippet
File Name PolarDB-for-PostgreSQL-2/_int_bool.c
Method bqarr_in(PG_FUNCTION_ARGS)

```
....
497.        state.str = NULL;
....
520.              ptr[i].type = state.str->type;
```

Use of Zero Initialized Pointer\Path 37:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=698
Status New

The variable declared in str at PolarDB-for-PostgreSQL-2/_int_bool.c in line 478 is not initialized when it is used by next at PolarDB-for-PostgreSQL-2/_int_bool.c in line 137.

	Source	Destination

File	PolarDB-for-PostgreSQL-2/_int_bool.c	PolarDB-for-PostgreSQL-2/_int_bool.c
Line	497	143
Object	str	next

Code Snippet
File Name PolarDB-for-PostgreSQL-2/_int_bool.c
Method bqarr_in(PG_FUNCTION_ARGS)

```
....
497.        state.str = NULL;
```

File Name PolarDB-for-PostgreSQL-2/_int_bool.c
Method pushquery(WORKSTATE *state, int32 type, int32 val)

```
....
143.        tmp->next = state->str;
```

Use of Zero Initialized Pointer\Path 38:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=699
Status	New

The variable declared in str at PolarDB-for-PostgreSQL-2/_int_bool.c in line 478 is not initialized when it is used by str at PolarDB-for-PostgreSQL-2/_int_bool.c in line 137.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/_int_bool.c	PolarDB-for-PostgreSQL-2/_int_bool.c
Line	497	144
Object	str	str

Code Snippet
File Name PolarDB-for-PostgreSQL-2/_int_bool.c
Method bqarr_in(PG_FUNCTION_ARGS)

```
....
497.        state.str = NULL;
```

File Name PolarDB-for-PostgreSQL-2/_int_bool.c
Method pushquery(WORKSTATE *state, int32 type, int32 val)

```
....
144.        state->str = tmp;
```

Use of Zero Initialized Pointer\Path 39:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=700
Status	New

The variable declared in rel at PolarDB-for-PostgreSQL-2/autoprewarm.c in line 439 is not initialized when it is used by rel at PolarDB-for-PostgreSQL-2/autoprewarm.c in line 439.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	491	503
Object	rel	rel

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method autoprewarm_database_main(Datum main_arg)

```
....
491.                    rel = NULL;
....
503.                    Assert(rel == NULL);
```

Use of Zero Initialized Pointer\Path 40:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=701
Status	New

The variable declared in wi_proc at PolarDB-for-PostgreSQL-2/autovacuum.c in line 1153 is not initialized when it is used by av_startingWorker at PolarDB-for-PostgreSQL-2/autovacuum.c in line 1153.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	1330	1333
Object	wi_proc	av_startingWorker

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method do_start_worker(void)

```
....
1330.             worker->wi_proc = NULL;
....
1333.             AutoVacuumShmem->av_startingWorker = worker;
```

Use of Zero Initialized Pointer\Path 41:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=702
Status	New

The variable declared in currentCommand at PolarDB-for-PostgreSQL-2/event_trigger.c in line 1249 is not initialized when it is used by currentEventTriggerState at PolarDB-for-PostgreSQL-2/event_trigger.c in line 1249.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/event_trigger.c	PolarDB-for-PostgreSQL-2/event_trigger.c
Line	1273	1276
Object	currentCommand	currentEventTriggerState

Code Snippet
File Name PolarDB-for-PostgreSQL-2/event_trigger.c
Method EventTriggerBeginCompleteQuery(void)

```
....
1273.       state->currentCommand = NULL;
....
1276.       currentEventTriggerState = state;
```

Use of Zero Initialized Pointer\Path 42:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=703
Status	New

The variable declared in parsetree at PolarDB-for-PostgreSQL-2/event_trigger.c in line 1838 is not initialized when it is used by currentEventTriggerState at PolarDB-for-PostgreSQL-2/event_trigger.c in line 1838.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/event_trigger.c	PolarDB-for-PostgreSQL-2/event_trigger.c
Line	1868	1870
Object	parsetree	currentEventTriggerState

Code Snippet
File Name PolarDB-for-PostgreSQL-2/event_trigger.c
Method EventTriggerCollectGrant(InternalGrant *istmt)

```
....
1868.       command->parsetree = NULL;
....
1870.       currentEventTriggerState->commandList =
```

Use of Zero Initialized Pointer\Path 43:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=704
Status	New

The variable declared in argnull at PolarDB-for-PostgreSQL-2/execExpr.c in line 673 is not initialized when it is used by resnull at PolarDB-for-PostgreSQL-2/execExpr.c in line 673.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c
Line	1992	1456
Object	argnull	resnull

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExecInitExprRec(Expr *node, ExprState *state,

```
....
1992.                                   scratch.d.xmlexpr.argnull = NULL;
....
1456.                                         scratch.resnull = resnull;
```

Use of Zero Initialized Pointer\Path 44:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=705
Status	New

The variable declared in argvalue at PolarDB-for-PostgreSQL-2/execExpr.c in line 673 is not initialized when it is used by resnull at PolarDB-for-PostgreSQL-2/execExpr.c in line 673.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c
Line	1991	1456
Object	argvalue	resnull

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExecInitExprRec(Expr *node, ExprState *state,

```
....
1991.                                   scratch.d.xmlexpr.argvalue = NULL;
....
1456.                                         scratch.resnull = resnull;
```

Use of Zero Initialized Pointer\Path 45:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=706
Status	New

The variable declared in named_argnull at PolarDB-for-PostgreSQL-2/execExpr.c in line 673 is not initialized when it is used by resnull at PolarDB-for-PostgreSQL-2/execExpr.c in line 673.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c
Line	1979	1456
Object	named_argnull	resnull

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExecInitExprRec(Expr *node, ExprState *state,

```
....
1979.                                   scratch.d.xmlexpr.named_argnull =
NULL;
....
1456.                                         scratch.resnull = resnull;
```

Use of Zero Initialized Pointer\Path 46:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=707
Status	New

The variable declared in named_argvalue at PolarDB-for-PostgreSQL-2/execExpr.c in line 673 is not initialized when it is used by resnull at PolarDB-for-PostgreSQL-2/execExpr.c in line 673.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c
Line	1978	1456
Object	named_argvalue	resnull

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExecInitExprRec(Expr *node, ExprState *state,

```
....
1978.                                   scratch.d.xmlexpr.named_argvalue =
NULL;
....
1456.                                         scratch.resnull = resnull;
```

Use of Zero Initialized Pointer\Path 47:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=708
Status	New

The variable declared in argnull at PolarDB-for-PostgreSQL-2/execExpr.c in line 673 is not initialized when it is used by resvalue at PolarDB-for-PostgreSQL-2/execExpr.c in line 673.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c
Line	1992	1455
Object	argnull	resvalue

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExecInitExprRec(Expr *node, ExprState *state,

```
....
1992.                                   scratch.d.xmlexpr.argnull = NULL;
....
1455.                                         scratch.resvalue = resv;
```

Use of Zero Initialized Pointer\Path 48:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=709
Status	New

The variable declared in argvalue at PolarDB-for-PostgreSQL-2/execExpr.c in line 673 is not initialized when it is used by resvalue at PolarDB-for-PostgreSQL-2/execExpr.c in line 673.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c
Line	1991	1455

Object	argvalue	resvalue

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExecInitExprRec(Expr *node, ExprState *state,

```
....
1991.                                   scratch.d.xmlexpr.argvalue = NULL;
....
1455.                                         scratch.resvalue = resv;
```

Use of Zero Initialized Pointer\Path 49:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=710
Status	New

The variable declared in named_argnull at PolarDB-for-PostgreSQL-2/execExpr.c in line 673 is not initialized when it is used by resvalue at PolarDB-for-PostgreSQL-2/execExpr.c in line 673.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c
Line	1979	1455
Object	named_argnull	resvalue

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExecInitExprRec(Expr *node, ExprState *state,

```
....
1979.                                   scratch.d.xmlexpr.named_argnull =
NULL;
....
1455.                                         scratch.resvalue = resv;
```

Use of Zero Initialized Pointer\Path 50:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=711
Status	New

The variable declared in named_argvalue at PolarDB-for-PostgreSQL-2/execExpr.c in line 673 is not initialized when it is used by resvalue at PolarDB-for-PostgreSQL-2/execExpr.c in line 673.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c

Line	1978	1455
Object	named_argvalue	resvalue

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExecInitExprRec(Expr *node, ExprState *state,

```
....
1978.                                   scratch.d.xmlexpr.named_argvalue =
NULL;
....
1455.                                         scratch.resvalue = resv;
```

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=1020028&projectid=20023&pathid=634
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	895	15458
Object	polar_enable_lazy_checkpoint	polar_enable_lazy_checkpoint

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method bool polar_enable_lazy_checkpoint;

```
....
895.  bool    polar_enable_lazy_checkpoint;
```

File Name PolarDB-for-PostgreSQL-2/guc.c
Method polar_check_enable_full_page_writes(bool *newval, void **extra, GucSource source)

```
....
15458.                  source != PGC_S_DEFAULT && *newval &&
polar_enable_lazy_checkpoint)
```

Use of Uninitialized Variable\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=635
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	118	3042
Object	autovacuum_vac_thresh	autovacuum_vac_thresh

Code Snippet

File Name PolarDB-for-PostgreSQL-2/autovacuum.c

Method int autovacuum_vac_thresh;

```
....
118.  int               autovacuum_vac_thresh;
```

File Name PolarDB-for-PostgreSQL-2/autovacuum.c

Method relation_needs_vacanalyze(Oid relid,

```
....
3042.             : autovacuum_vac_thresh;
```

Use of Uninitialized Variable\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=636
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	120	3050
Object	autovacuum_anl_thresh	autovacuum_anl_thresh

Code Snippet

File Name PolarDB-for-PostgreSQL-2/autovacuum.c

Method int autovacuum_anl_thresh;

```
....
120.  int               autovacuum_anl_thresh;
```

File Name PolarDB-for-PostgreSQL-2/autovacuum.c

Method relation_needs_vacanalyze(Oid relid,

```
....
3050.              : autovacuum_anl_thresh;
```

Use of Uninitialized Variable\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=637
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	125	1827
Object	autovacuum_vac_cost_delay	autovacuum_vac_cost_delay

Code Snippet

File Name PolarDB-for-PostgreSQL-2/autovacuum.c

Method int autovacuum_vac_cost_delay;

```
....
125.  int                 autovacuum_vac_cost_delay;
```

File Name PolarDB-for-PostgreSQL-2/autovacuum.c

Method autovac_balance_cost(void)

```
....
1827.
autovacuum_vac_cost_delay : VacuumCostDelay);
```

Use of Uninitialized Variable\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=638
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	125	1826
Object	autovacuum_vac_cost_delay	autovacuum_vac_cost_delay

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method int autovacuum_vac_cost_delay;

```
....
125.  int               autovacuum_vac_cost_delay;
```

File Name PolarDB-for-PostgreSQL-2/autovacuum.c

Method autovac_balance_cost(void)

```
....
1826.       int               vac_cost_delay =
(autovacuum_vac_cost_delay >= 0 ?
```

Use of Uninitialized Variable\Path 6:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=639
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	125	2886
Object	autovacuum_vac_cost_delay	autovacuum_vac_cost_delay

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method int autovacuum_vac_cost_delay;

```
....
125.  int               autovacuum_vac_cost_delay;
```

File Name PolarDB-for-PostgreSQL-2/autovacuum.c

Method table_recheck_autovac(Oid relid, HTAB *table_toast_map,

```
....
2886.                   : (autovacuum_vac_cost_delay >= 0)
```

Use of Uninitialized Variable\Path 7:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=640

Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	125	2887
Object	autovacuum_vac_cost_delay	autovacuum_vac_cost_delay

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method int autovacuum_vac_cost_delay;

```
....
125.  int               autovacuum_vac_cost_delay;
```

File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method table_recheck_autovac(Oid relid, HTAB *table_toast_map,

```
....
2887.                   ? autovacuum_vac_cost_delay
```

Use of Uninitialized Variable\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=641
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	126	1825
Object	autovacuum_vac_cost_limit	autovacuum_vac_cost_limit

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method int autovacuum_vac_cost_limit;

```
....
126.  int               autovacuum_vac_cost_limit;
```

File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method autovac_balance_cost(void)

```
....
1825.
autovacuum_vac_cost_limit : VacuumCostLimit);
```

Use of Uninitialized Variable\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=642
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	126	1824
Object	autovacuum_vac_cost_limit	autovacuum_vac_cost_limit

Code Snippet

File Name PolarDB-for-PostgreSQL-2/autovacuum.c

Method int autovacuum_vac_cost_limit;

```
....
126.  int               autovacuum_vac_cost_limit;
```

File Name PolarDB-for-PostgreSQL-2/autovacuum.c

Method autovac_balance_cost(void)

```
....
1824.       int               vac_cost_limit =
(autovacuum_vac_cost_limit > 0 ?
```

Use of Uninitialized Variable\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=643
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	126	2893
Object	autovacuum_vac_cost_limit	autovacuum_vac_cost_limit

Code Snippet

File Name PolarDB-for-PostgreSQL-2/autovacuum.c

Method int autovacuum_vac_cost_limit;

```
....
126.  int               autovacuum_vac_cost_limit;
```

File Name PolarDB-for-PostgreSQL-2/autovacuum.c

Method table_recheck_autovac(Oid relid, HTAB *table_toast_map,

```
....
2893.                   : (autovacuum_vac_cost_limit > 0)
```

Use of Uninitialized Variable\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=644
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	126	2894
Object	autovacuum_vac_cost_limit	autovacuum_vac_cost_limit

Code Snippet

File Name PolarDB-for-PostgreSQL-2/autovacuum.c

Method int autovacuum_vac_cost_limit;

```
....
126.  int               autovacuum_vac_cost_limit;
```

File Name PolarDB-for-PostgreSQL-2/autovacuum.c

Method table_recheck_autovac(Oid relid, HTAB *table_toast_map,

```
....
2894.                   ? autovacuum_vac_cost_limit
```

Use of Uninitialized Variable\Path 12:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=645
Status	New

Source	Destination

File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	183	12265
Object	polar_max_non_super_conns	polar_max_non_super_conns

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method int polar_max_non_super_conns;

```
....
183.  int                     polar_max_non_super_conns;
```

File Name PolarDB-for-PostgreSQL-2/guc.c
Method polar_show_max_connections(void)

```
....
12265.            result = (polar_max_non_super_conns < 0 ||
(MaxConnections < polar_max_non_super_conns)) ?
```

Use of Uninitialized Variable\Path 13:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=646
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	183	12265
Object	polar_max_non_super_conns	polar_max_non_super_conns

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method int polar_max_non_super_conns;

```
....
183.  int                     polar_max_non_super_conns;
```

File Name PolarDB-for-PostgreSQL-2/guc.c
Method polar_show_max_connections(void)

```
....
12265.            result = (polar_max_non_super_conns < 0 ||
(MaxConnections < polar_max_non_super_conns)) ?
```

Use of Uninitialized Variable\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=647
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	183	12266
Object	polar_max_non_super_conns	polar_max_non_super_conns

Code Snippet

File Name PolarDB-for-PostgreSQL-2/guc.c

Method int polar_max_non_super_conns;

```
....
183.  int                     polar_max_non_super_conns;
```

File Name PolarDB-for-PostgreSQL-2/guc.c

Method polar_show_max_connections(void)

```
....
12266.                                MaxConnections :
polar_max_non_super_conns;
```

Use of Uninitialized Variable\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=648
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	210	15395
Object	polar_nblocks_cache_mode	polar_nblocks_cache_mode

Code Snippet

File Name PolarDB-for-PostgreSQL-2/guc.c

Method int polar_nblocks_cache_mode;

```
....
210.  int               polar_nblocks_cache_mode;
```

File Name PolarDB-for-PostgreSQL-2/guc.c

Method polar_check_nblocks_cache_mode(int *newval, void **extra, GucSource source)

```
....
15395.            (polar_nblocks_cache_mode ==
POLAR_NBLOCKS_CACHE_OFF_MODE) &&
```

Use of Uninitialized Variable\Path 16:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=649
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	230	15435
Object	polar_crash_recovery_rto	polar_crash_recovery_rto

Code Snippet

File Name PolarDB-for-PostgreSQL-2/guc.c

Method int polar_crash_recovery_rto;

```
....
230.  int         polar_crash_recovery_rto;
```

File Name PolarDB-for-PostgreSQL-2/guc.c

Method polar_assign_crash_recovery_rto_delay_time(const int newval, void *extra)

```
....
15435.
polar_crash_recovery_rto != 0);
```

Use of Uninitialized Variable\Path 17:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=650
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	230	15423

Object	polar_crash_recovery_rto	polar_crash_recovery_rto

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method int polar_crash_recovery_rto;

```
....
230.  int         polar_crash_recovery_rto;
```

File Name PolarDB-for-PostgreSQL-2/guc.c

Method polar_assign_crash_recovery_rto_threshold(double newval, void *extra)

```
....
15423.
polar_crash_recovery_rto != 0 &&
```

Use of Uninitialized Variable\Path 18:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=651
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	231	15424
Object	polar_crash_recovery_rto_delay_count	polar_crash_recovery_rto_delay_count

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method int polar_crash_recovery_rto_delay_count;

```
....
231.  int         polar_crash_recovery_rto_delay_count;
```

File Name PolarDB-for-PostgreSQL-2/guc.c

Method polar_assign_crash_recovery_rto_threshold(double newval, void *extra)

```
....
15424.
polar_crash_recovery_rto_delay_count != 0);
```

Use of Uninitialized Variable\Path 19:

Severity Medium
Result State To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=652
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	231	15413
Object	polar_crash_recovery_rto_delay_count	polar_crash_recovery_rto_delay_count

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method int polar_crash_recovery_rto_delay_count;

```
....
231.  int         polar_crash_recovery_rto_delay_count;
```

File Name PolarDB-for-PostgreSQL-2/guc.c
Method polar_assign_crash_recovery_rto(const int newval, void *extra)

```
....
15413.
polar_crash_recovery_rto_delay_count != 0);
```

Use of Uninitialized Variable\Path 20:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=653
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuum.c	PolarDB-for-PostgreSQL-2/vacuum.c
Line	61	646
Object	vacuum_freeze_min_age	vacuum_freeze_min_age

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuum.c
Method int vacuum_freeze_min_age;

```
....
61.  int               vacuum_freeze_min_age;
```

File Name PolarDB-for-PostgreSQL-2/vacuum.c

Method vacuum_set_xid_limits(Relation rel,

```
....
646.              freezemin = vacuum_freeze_min_age;
```

Use of Uninitialized Variable\Path 21:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=654
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuum.c	PolarDB-for-PostgreSQL-2/vacuum.c
Line	62	737
Object	vacuum_freeze_table_age	vacuum_freeze_table_age

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuum.c
Method int vacuum_freeze_table_age;

```
....
62.  int                vacuum_freeze_table_age;
```

File Name PolarDB-for-PostgreSQL-2/vacuum.c
Method vacuum_set_xid_limits(Relation rel,

```
....
737.                    freezetable = vacuum_freeze_table_age;
```

Use of Uninitialized Variable\Path 22:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=655
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuum.c	PolarDB-for-PostgreSQL-2/vacuum.c
Line	63	692
Object	vacuum_multixact_freeze_min_age	vacuum_multixact_freeze_min_age

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuum.c

Method	int	vacuum_multixact_freeze_min_age;

```
....
63.  int                vacuum_multixact_freeze_min_age;
```

File Name	PolarDB-for-PostgreSQL-2/vacuum.c
Method	vacuum_set_xid_limits(Relation rel,

```
....
692.              mxid_freezemin = vacuum_multixact_freeze_min_age;
```

Use of Uninitialized Variable\Path 23:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=656
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuum.c	PolarDB-for-PostgreSQL-2/vacuum.c
Line	64	761
Object	vacuum_multixact_freeze_table_age	vacuum_multixact_freeze_table_age

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/vacuum.c	
Method	int	vacuum_multixact_freeze_table_age;

```
....
64.  int                vacuum_multixact_freeze_table_age;
```

File Name	PolarDB-for-PostgreSQL-2/vacuum.c
Method	vacuum_set_xid_limits(Relation rel,

```
....
761.                    freezetable = vacuum_multixact_freeze_table_age;
```

Use of Uninitialized Variable\Path 24:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=657
Status	New

	Source	Destination

File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	119	3038
Object	autovacuum_vac_scale	autovacuum_vac_scale

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method double autovacuum_vac_scale;

```
....
119.  double            autovacuum_vac_scale;
```

File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method relation_needs_vacanalyze(Oid relid,

```
....
3038.             : autovacuum_vac_scale;
```

Use of Uninitialized Variable\Path 25:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=658
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	121	3046
Object	autovacuum_anl_scale	autovacuum_anl_scale

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method double autovacuum_anl_scale;

```
....
121.  double            autovacuum_anl_scale;
```

File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method relation_needs_vacanalyze(Oid relid,

```
....
3046.             : autovacuum_anl_scale;
```

Use of Uninitialized Variable\Path 26:

Severity Medium

Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=659
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	229	15434
Object	polar_crash_recovery_rto_threshold	polar_crash_recovery_rto_threshold

Code Snippet

File Name PolarDB-for-PostgreSQL-2/guc.c

Method double polar_crash_recovery_rto_threshold;

```
....
229.  double            polar_crash_recovery_rto_threshold;
```

File Name PolarDB-for-PostgreSQL-2/guc.c

Method polar_assign_crash_recovery_rto_delay_time(const int newval, void *extra)

```
....
15434.
polar_crash_recovery_rto_threshold != 0 &&
```

Use of Uninitialized Variable\Path 27:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=660
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	229	15412
Object	polar_crash_recovery_rto_threshold	polar_crash_recovery_rto_threshold

Code Snippet

File Name PolarDB-for-PostgreSQL-2/guc.c

Method double polar_crash_recovery_rto_threshold;

```
....
229.  double            polar_crash_recovery_rto_threshold;
```

File Name PolarDB-for-PostgreSQL-2/guc.c

Method	polar_assign_crash_recovery_rto(const int newval, void *extra)

```
....
15412.
polar_crash_recovery_rto_threshold != 0 &&
```

Use of Uninitialized Variable\Path 28:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=661
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/array_userfuncs.c	PolarDB-for-PostgreSQL-2/array_userfuncs.c
Line	162	205
Object	lb0	lb0

Code Snippet
File Name PolarDB-for-PostgreSQL-2/array_userfuncs.c
Method array_prepend(PG_FUNCTION_ARGS)

```
....
162.        int                lb0;
....
205.                eah->lbound[0] = lb0;
```

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=1020028&projectid=20023&pathid=240
Status	New

Calling free() (line 153) on a variable that was not dynamically allocated (line 153) in file PolarDB-for-PostgreSQL-2/fe-secure-common.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/fe-secure-common.c	PolarDB-for-PostgreSQL-2/fe-secure-common.c
Line	208	208
Object	first_name	first_name

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/fe-secure-common.c
Method	pq_verify_peer_name_matches_certificate(PGconn *conn)

```
....
208.                free(first_name);
```

MemoryFree on StackVariable\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=241
Status	New

Calling free() (line 8218) on a variable that was not dynamically allocated (line 8218) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8227	8227
Object	oldval	oldval

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/guc.c
Method	set_string_field(struct config_string *conf, char **field, char *newval)

```
....
8227.               free(oldval);
```

MemoryFree on StackVariable\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=242
Status	New

Calling free() (line 8279) on a variable that was not dynamically allocated (line 8279) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8288	8288
Object	oldval	oldval

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/guc.c

Method	set_extra_field(struct config_generic *gconf, void **field, void *newval)

```
....
8288.               free(oldval);
```

MemoryFree on StackVariable\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=243
Status	New

Calling free() (line 8947) on a variable that was not dynamically allocated (line 8947) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	9030	9030
Object	configdir	configdir

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/guc.c
Method	SelectConfigFiles(const char *userDoption, const char *progname)

```
....
9030.               free(configdir);
```

MemoryFree on StackVariable\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=244
Status	New

Calling free() (line 8947) on a variable that was not dynamically allocated (line 8947) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	9135	9135
Object	configdir	configdir

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/guc.c
Method	SelectConfigFiles(const char *userDoption, const char *progname)

```
....
9135.       free(configdir);
```

MemoryFree on StackVariable\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=245
Status	New

Calling free() (line 10194) on a variable that was not dynamically allocated (line 10194) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	10524	10524
Object	newextra	newextra

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method set_config_option(const char *name, const char *value,

```
....
10524.                                  free(newextra);
```

MemoryFree on StackVariable\Path 7:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=246
Status	New

Calling free() (line 10194) on a variable that was not dynamically allocated (line 10194) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	10614	10614
Object	newextra	newextra

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method set_config_option(const char *name, const char *value,

```
....
10614.                                    free(newextra);
```

MemoryFree on StackVariable\Path 8:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=247
Status	New

Calling free() (line 10194) on a variable that was not dynamically allocated (line 10194) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	10704	10704
Object	newextra	newextra

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method set_config_option(const char *name, const char *value,

```
....
10704.                                    free(newextra);
```

MemoryFree on StackVariable\Path 9:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=248
Status	New

Calling free() (line 10194) on a variable that was not dynamically allocated (line 10194) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	10817	10817
Object	newextra	newextra

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method set_config_option(const char *name, const char *value,

```
....
10817.                                  free(newextra);
```

MemoryFree on StackVariable\Path 10:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=249
Status	New

Calling free() (line 10194) on a variable that was not dynamically allocated (line 10194) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	10907	10907
Object	newextra	newextra

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method set_config_option(const char *name, const char *value,

```
....
10907.                                  free(newextra);
```

MemoryFree on StackVariable\Path 11:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=250
Status	New

Calling free() (line 11240) on a variable that was not dynamically allocated (line 11240) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	11278	11278
Object	escaped	escaped

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_auto_conf_file(int fd, const char *filename, ConfigVariable *head)

```
....
11278.            free(escaped);
```

MemoryFree on StackVariable\Path 12:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=251
Status	New

Calling free() (line 11378) on a variable that was not dynamically allocated (line 11378) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	11467	11467
Object	newextra	newextra

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method AlterSystemSetConfigFile(AlterSystemStmt *altersysstmt)

```
....
11467.                          free(newextra);
```

MemoryFree on StackVariable\Path 13:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=252
Status	New

Calling free() (line 11864) on a variable that was not dynamically allocated (line 11864) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	11953	11953
Object	pHolder	pHolder

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method define_custom_variable(struct config_generic *variable)

```
....
11953.      free(pHolder);
```

MemoryFree on StackVariable\Path 14:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=253
Status	New

Calling free() (line 13235) on a variable that was not dynamically allocated (line 13235) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13293	13293
Object	varname	varname

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method read_nondefault_variables(void)

```
....
13293.            free(varname);
```

MemoryFree on StackVariable\Path 15:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=254
Status	New

Calling free() (line 13235) on a variable that was not dynamically allocated (line 13235) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13294	13294
Object	varvalue	varvalue

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method read_nondefault_variables(void)

```
....
13294.            free(varvalue);
```

MemoryFree on StackVariable\Path 16:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=255
Status	New

Calling free() (line 13235) on a variable that was not dynamically allocated (line 13235) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13295	13295
Object	varsourcefile	varsourcefile

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method read_nondefault_variables(void)

```
....
13295.            free(varsourcefile);
```

MemoryFree on StackVariable\Path 17:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=256
Status	New

Calling free() (line 15328) on a variable that was not dynamically allocated (line 15328) in file PolarDB-for-PostgreSQL-2/guc.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	15353	15353
Object	newextra	newextra

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method polar_parameter_check_value_internal(const char* guc_name, const char* guc_value)

```
....
15353.            free(newextra);
```

MemoryFree on StackVariable\Path 18:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=257
Status	New

Calling free() (line 3658) on a variable that was not dynamically allocated (line 3658) in file PolarDB-for-PostgreSQL-2/postgres.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	3854	3854
Object	name	name

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres.c
Method process_postgres_switches(int argc, char *argv[], GucContext ctx,

```
....
3854.                                 free(name);
```

MemoryFree on StackVariable\Path 19:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=258
Status	New

Calling free() (line 3658) on a variable that was not dynamically allocated (line 3658) in file PolarDB-for-PostgreSQL-2/postgres.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	3856	3856
Object	value	value

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres.c
Method process_postgres_switches(int argc, char *argv[], GucContext ctx,

```
....
3856.                                   free(value);
```

MemoryFree on StackVariable\Path 20:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=259
Status	New

Calling free() (line 816) on a variable that was not dynamically allocated (line 816) in file PolarDB-for-PostgreSQL-2/timestamp.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/timestamp.c	PolarDB-for-PostgreSQL-2/timestamp.c
Line	852	852
Object	mstr	mstr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/timestamp.c
Method PGTYPEStimestamp_defmt_asc(const char *str, const char *fmt, timestamp * d)

```
....
852.        free(mstr);
```

MemoryFree on StackVariable\Path 21:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=260
Status	New

Calling free() (line 816) on a variable that was not dynamically allocated (line 816) in file PolarDB-for-PostgreSQL-2/timestamp.c may result with a crash.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/timestamp.c	PolarDB-for-PostgreSQL-2/timestamp.c
Line	853	853
Object	mfmt	mfmt

Code Snippet
File Name PolarDB-for-PostgreSQL-2/timestamp.c
Method PGTYPEStimestamp_defmt_asc(const char *str, const char *fmt, timestamp * d)

```
....
853.         free(mfmt);
```

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=1020028&projectid=20023&pathid=626
Status	New

The variable declared in newquery at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 3197 is not initialized when it is used by newquery at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 3197.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeFuncs.c	PolarDB-for-PostgreSQL-2/nodeFuncs.c
Line	3206	3209
Object	newquery	newquery

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeFuncs.c
Method query_tree_mutator(Query *query,

```
....
3206.             Query    *newquery;
....
3209.             query = newquery;
```

Use of Uninitialized Pointer\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=627
Status	New

The variable declared in newnode at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 2515 is not initialized when it is used by newnode at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 2515.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeFuncs.c	PolarDB-for-PostgreSQL-2/nodeFuncs.c

Line	2590	2600
Object	newnode	newnode

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeFuncs.c
Method expression_tree_mutator(Node *node,

```
....
2590.                           Aggref          *newnode;
....
2600.                           return (Node *) newnode;
```

Use of Uninitialized Pointer\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=628
Status	New

The variable declared in newnode at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 2515 is not initialized when it is used by aggargtypes at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 2515.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeFuncs.c	PolarDB-for-PostgreSQL-2/nodeFuncs.c
Line	2590	2594
Object	newnode	aggargtypes

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeFuncs.c
Method expression_tree_mutator(Node *node,

```
....
2590.                           Aggref          *newnode;
....
2594.                           newnode->aggargtypes = list_copy(aggref-
>aggargtypes);
```

Use of Uninitialized Pointer\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=629
Status	New

The variable declared in newnode at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 2515 is not initialized when it is used by cols at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 2515.

	Source	Destination

File	PolarDB-for-PostgreSQL-2/nodeFuncs.c	PolarDB-for-PostgreSQL-2/nodeFuncs.c
Line	2606	2621
Object	newnode	cols

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeFuncs.c
Method expression_tree_mutator(Node *node,

```
....
2606.                           GroupingFunc *newnode;
....
2621.                           newnode->cols = list_copy(grouping->cols);
```

Use of Uninitialized Pointer\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=630
Status	New

The variable declared in newnode at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 2515 is not initialized when it is used by refs at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 2515.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeFuncs.c	PolarDB-for-PostgreSQL-2/nodeFuncs.c
Line	2606	2620
Object	newnode	refs

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeFuncs.c
Method expression_tree_mutator(Node *node,

```
....
2606.                           GroupingFunc *newnode;
....
2620.                           newnode->refs = list_copy(grouping->refs);
```

Use of Uninitialized Pointer\Path 6:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=631
Status	New

The variable declared in newnode at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 2515 is not initialized when it is used by newnode at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 2515.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeFuncs.c	PolarDB-for-PostgreSQL-2/nodeFuncs.c
Line	2606	2623
Object	newnode	newnode

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeFuncs.c
Method expression_tree_mutator(Node *node,

```
....
2606.                           GroupingFunc *newnode;
....
2623.                           return (Node *) newnode;
```

Use of Uninitialized Pointer\Path 7:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=632
Status New

The variable declared in newnode at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 2515 is not initialized when it is used by newnode at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 2515.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeFuncs.c	PolarDB-for-PostgreSQL-2/nodeFuncs.c
Line	2777	2783
Object	newnode	newnode

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeFuncs.c
Method expression_tree_mutator(Node *node,

```
....
2777.                           FieldStore *newnode;
....
2783.                           return (Node *) newnode;
```

Use of Uninitialized Pointer\Path 8:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=633
Status New

The variable declared in newnode at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 2515 is not initialized when it is used by fieldnums at PolarDB-for-PostgreSQL-2/nodeFuncs.c in line 2515.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeFuncs.c	PolarDB-for-PostgreSQL-2/nodeFuncs.c
Line	2777	2782
Object	newnode	fieldnums

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeFuncs.c
Method expression_tree_mutator(Node *node,

```
....
2777.                           FieldStore *newnode;
....
2782.                           newnode->fieldnums = list_copy(fstore-
>fieldnums);
```

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=1020028&projectid=20023&pathid=45
Status	New

The application performs an illegal operation in autovac_balance_cost, in PolarDB-for-PostgreSQL-2/autovacuum.c. In line 1814, the program attempts to divide by vac_cost_delay, 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 vac_cost_delay in autovac_balance_cost of PolarDB-for-PostgreSQL-2/autovacuum.c, at line 1814.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	1857	1857
Object	vac_cost_delay	vac_cost_delay

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method autovac_balance_cost(void)

```
....
1857.       cost_avail = (double) vac_cost_limit / vac_cost_delay;
```

Divide By Zero\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=200

	23&pathid=46
Status	New

The application performs an illegal operation in autovac_balance_cost, in PolarDB-for-PostgreSQL-2/autovacuum.c. In line 1814, the program attempts to divide by cost_total, 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 cost_total in autovac_balance_cost of PolarDB-for-PostgreSQL-2/autovacuum.c, at line 1814.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	1867	1867
Object	cost_total	cost_total

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method autovac_balance_cost(void)

```
....
1867.                   (cost_avail * worker->wi_cost_limit_base /
cost_total);
```

Divide By Zero\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=47
Status	New

The application performs an illegal operation in circle_poly, in PolarDB-for-PostgreSQL-2/geo_ops.c. In line 5155, the program attempts to divide by npts, 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 npts in circle_poly of PolarDB-for-PostgreSQL-2/geo_ops.c, at line 5155.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/geo_ops.c	PolarDB-for-PostgreSQL-2/geo_ops.c
Line	5189	5189
Object	npts	npts

Code Snippet
File Name PolarDB-for-PostgreSQL-2/geo_ops.c
Method circle_poly(PG_FUNCTION_ARGS)

```
....
5189.       anglestep = (2.0 * M_PI) / npts;
```

Divide By Zero\Path 4:

Severity	Medium
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=48
Status	New

The application performs an illegal operation in vac_estimate_reltuples, in PolarDB-for-PostgreSQL-2/vacuum.c. In line 796, the program attempts to divide by scanned_pages, 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 scanned_pages in vac_estimate_reltuples of PolarDB-for-PostgreSQL-2/vacuum.c, at line 796.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuum.c	PolarDB-for-PostgreSQL-2/vacuum.c
Line	825	825
Object	scanned_pages	scanned_pages

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuum.c
Method vac_estimate_reltuples(Relation relation,

```
....
825.              return floor((scanned_tuples / scanned_pages) *
total_pages + 0.5);
```

Divide By Zero\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=49
Status	New

The application performs an illegal operation in vac_estimate_reltuples, in PolarDB-for-PostgreSQL-2/vacuum.c. In line 796, the program attempts to divide by old_rel_pages, 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 old_rel_pages in vac_estimate_reltuples of PolarDB-for-PostgreSQL-2/vacuum.c, at line 796.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuum.c	PolarDB-for-PostgreSQL-2/vacuum.c
Line	833	833
Object	old_rel_pages	old_rel_pages

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuum.c
Method vac_estimate_reltuples(Relation relation,

```
....
833.        old_density = old_rel_tuples / old_rel_pages;
```

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=1020028&projectid=20023&pathid=278
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 73 of PolarDB-for-PostgreSQL-2/pg_operator.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/pg_operator.c	PolarDB-for-PostgreSQL-2/pg_operator.c
Line	103	103
Object	AssignExpr	AssignExpr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/pg_operator.c
Method validOperatorName(const char *name)

```
....
103.              for (ic = len - 2; ic >= 0; ic--)
```

Integer Overflow\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=279
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 5075 of PolarDB-for-PostgreSQL-2/postgres.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	5090	5090
Object	AssignExpr	AssignExpr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres.c

Method	log_disconnections(int code, Datum arg)

```
....
5090.       hours = secs / SECS_PER_HOUR;
```

Integer Overflow\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=280
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 5075 of PolarDB-for-PostgreSQL-2/postgres.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	5092	5092
Object	AssignExpr	AssignExpr

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/postgres.c
Method	log_disconnections(int code, Datum arg)

```
....
5092.       minutes = secs / SECS_PER_MINUTE;
```

Integer Overflow\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=281
Status	New

A variable of a larger data type, AssignExpr, is being assigned to a smaller data type, in 5075 of PolarDB-for-PostgreSQL-2/postgres.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	5093	5093
Object	AssignExpr	AssignExpr

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/postgres.c
Method	log_disconnections(int code, Datum arg)

```
....
5093.       seconds = secs % SECS_PER_MINUTE;
```

Integer Overflow\Path 5:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=282
Status	New

A variable of a larger data type, limit, is being assigned to a smaller data type, in 1814 of PolarDB-for-PostgreSQL-2/autovacuum.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	1866	1866
Object	limit	limit

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method autovac_balance_cost(void)

```
....
1866.                   int               limit = (int)
```

Missing Precision

Query Path:
CPP\Cx\CPP Buffer Overflow\Missing Precision Version:0

Categories

NIST SP 800-53: SI-10 Information Input Validation (P1)

Description

Missing Precision\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=264
Status	New

The size of the buffer used by SelectConfigFiles in configdir, at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that SelectConfigFiles passes to getenv, at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c

Line	8957	9121
Object	getenv	configdir

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method SelectConfigFiles(const char *userDoption, const char *progname)

```
....
8957.            configdir = make_absolute_path(getenv("PGDATA"));
....
9121.            sprintf(fname, "%s/%s", configdir, IDENT_FILENAME);
```

Missing Precision\Path 2:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=265
Status New

The size of the buffer used by SelectConfigFiles in configdir, at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that SelectConfigFiles passes to getenv, at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8957	9098
Object	getenv	configdir

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method SelectConfigFiles(const char *userDoption, const char *progname)

```
....
8957.            configdir = make_absolute_path(getenv("PGDATA"));
....
9098.            sprintf(fname, "%s/%s", configdir, HBA_FILENAME);
```

Missing Precision\Path 3:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=266
Status New

The size of the buffer used by SelectConfigFiles in configdir, at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that SelectConfigFiles passes to getenv, at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8957	9009
Object	getenv	configdir

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method SelectConfigFiles(const char *userDoption, const char *progname)

```
....
8957.             configdir = make_absolute_path(getenv("PGDATA"));
....
9009.             sprintf(fname, "%s/%s", configdir,
POLAR_DMA_FILENAME);
```

Missing Precision\Path 4:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=267
Status	New

The size of the buffer used by SelectConfigFiles in configdir, at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that SelectConfigFiles passes to getenv, at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8957	8982
Object	getenv	configdir

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method SelectConfigFiles(const char *userDoption, const char *progname)

```
....
8957.             configdir = make_absolute_path(getenv("PGDATA"));
....
8982.             sprintf(fname, "%s/%s", configdir, CONFIG_FILENAME);
```

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=1020028&projectid=20023&pathid=620
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copyfuncs.c	PolarDB-for-PostgreSQL-2/copyfuncs.c
Line	4964	4964
Object	neW	neW

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copyfuncs.c
Method _copyList(const List *from)

```
....
4964.       List     *new;
```

Memory Leak\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=621
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/pl_handler.c	PolarDB-for-PostgreSQL-2/pl_handler.c
Line	115	115
Object	myextra	myextra

Code Snippet
File Name PolarDB-for-PostgreSQL-2/pl_handler.c
Method plpgsql_extra_checks_check_hook(char **newvalue, void **extra, GucSource source)

```
....
115.        myextra = (int *) malloc(sizeof(int));
```

Memory Leak\Path 3:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=622
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	3717	3717
Object	userDoption	userDoption

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres.c
Method process_postgres_switches(int argc, char *argv[], GucContext ctx,

```
....
3717.                                   userDoption = strdup(optarg);
```

Memory Leak\Path 4:

Severity Medium
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=623
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	3873	3873
Object	dbname	dbname

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres.c
Method process_postgres_switches(int argc, char *argv[], GucContext ctx,

```
....
3873.                *dbname = strdup(argv[optind++]);
```

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=1020028&projectid=20023&pathid=262
Status New

The function size in PolarDB-for-PostgreSQL-2/guc.c at line 8146 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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8153	8153
Object	size	size

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method guc_malloc(int elevel, size_t size)

```
....
8153.       data = malloc(size);
```

Wrong Size t Allocation\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=263
Status	New

The function namelen in PolarDB-for-PostgreSQL-2/fe-secure-common.c at line 85 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	PolarDB-for-PostgreSQL-2/fe-secure-common.c	PolarDB-for-PostgreSQL-2/fe-secure-common.c
Line	106	106
Object	namelen	namelen

Code Snippet
File Name PolarDB-for-PostgreSQL-2/fe-secure-common.c
Method pq_verify_peer_name_matches_certificate_name(PGconn *conn,

```
....
106.        name = malloc(namelen + 1);
```

Use After Free

Query Path:
CPP\Cx\CPP Medium Threat\Use After Free Version:1

Categories

NIST SP 800-53: SC-5 Denial of Service Protection (P1)
OWASP Top 10 2017: A1-Injection

Description

Use After Free\Path 1:

Severity	Medium
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=624
Status	New

The pointer escaped at PolarDB-for-PostgreSQL-2/guc.c in line 11240 is being used after it has been freed.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	11278	11277
Object	escaped	escaped

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_auto_conf_file(int fd, const char *filename, ConfigVariable *head)

```
....
11278.            free(escaped);
....
11277.            appendStringInfoString(&buf, escaped);
```

Use After Free\Path 2:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=625
Status	New

The pointer name at PolarDB-for-PostgreSQL-2/postgres.c in line 3658 is being used after it has been freed.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	3854	3853
Object	name	name

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres.c
Method process_postgres_switches(int argc, char *argv[], GucContext ctx,

```
....
3854.                                   free(name);
....
3853.                                   SetConfigOption(name, value, ctx,
gucsource);
```

Off by One Error in Methods

Query Path:
CPP\Cx\CPP Buffer Overflow\Off by One Error in Methods Version:0

Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.2 - Buffer overflows
NIST SP 800-53: SI-16 Memory Protection (P1)
OWASP Top 10 2017: A1-Injection

Description

Off by One Error in Methods\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=261
Status	New

The buffer allocated by sizeof in PolarDB-for-PostgreSQL-2/misc.c at line 506 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/misc.c	PolarDB-for-PostgreSQL-2/misc.c
Line	543	543
Object	targetpath	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/misc.c
Method pg_tablespace_location(PG_FUNCTION_ARGS)

```
....
543.        rllen = readlink(sourcepath, targetpath,
sizeof(targetpath));
```

Char Overflow

Query Path:
CPP\Cx\CPP Integer Overflow\Char Overflow Version:1

Categories

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

Description

Char Overflow\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=277
Status	New

A variable of a larger data type, nthbit, is being assigned to a smaller data type, in 319 of PolarDB-for-PostgreSQL-2/px_disp_query.c. This will cause a loss of data, often the significant bits of a numerical value or the sign bit.

	Source	Destination

File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	322	322
Object	nthbit	nthbit

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method mark_bit(char *bits, int nth)

```
....
322.        char        nthbit = 1 << (nth & 7);
```

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=1020028&projectid=20023&pathid=618
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13824	13824
Object	name	name

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method ProcessGUCArray(ArrayType *array,

```
....
13824.            free(name);
```

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=1020028&projectid=20023&pathid=619
Status	New

Method main at line 96 of PolarDB-for-PostgreSQL-2/vacuumdb.c defines prompt_password, which is designated to contain user passwords. However, while plaintext passwords are later assigned to prompt_password, this variable is never cleared from memory.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuumdb.c	PolarDB-for-PostgreSQL-2/vacuumdb.c
Line	128	128
Object	prompt_password	prompt_password

Code Snippet

File Name PolarDB-for-PostgreSQL-2/vacuumdb.c

Method main(int argc, char *argv[])

```
....
128.        enum trivalue prompt_password = TRI_DEFAULT;
```

Uncontrolled Recursion

Query Path:

CPP\Cx\CPP Medium Threat\Uncontrolled Recursion Version:1

Description

Uncontrolled Recursion\Path 1:

Severity	Medium
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=841
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	373	373
Object	markbit_dep_children	markbit_dep_children

Code Snippet

File Name PolarDB-for-PostgreSQL-2/px_disp_query.c

Method markbit_dep_children(SliceTable * sliceTable, int sliceIdx,

```
....
373.                markbit_dep_children(sliceTable, childIndex,
```

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=1020028&projectid=20023&pathid=843
Status	New

The infix method calls the sprintf function, at line 565 of PolarDB-for-PostgreSQL-2/_int_bool.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	PolarDB-for-PostgreSQL-2/_int_bool.c	PolarDB-for-PostgreSQL-2/_int_bool.c
Line	573	573
Object	sprintf	sprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/_int_bool.c
Method infix(INFIX *in, bool first)

```
....
573.                sprintf(in->cur, "%d", in->curpol->val);
```

Unchecked Return Value\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=844
Status	New

The infix method calls the sprintf function, at line 565 of PolarDB-for-PostgreSQL-2/_int_bool.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	PolarDB-for-PostgreSQL-2/_int_bool.c	PolarDB-for-PostgreSQL-2/_int_bool.c
Line	590	590
Object	sprintf	sprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/_int_bool.c
Method	infix(INFIX *in, bool first)

```
....
590.                    sprintf(in->cur, "( ");
```

Unchecked Return Value\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=845
Status	New

The infix method calls the sprintf function, at line 565 of PolarDB-for-PostgreSQL-2/_int_bool.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	PolarDB-for-PostgreSQL-2/_int_bool.c	PolarDB-for-PostgreSQL-2/_int_bool.c
Line	597	597
Object	sprintf	sprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/_int_bool.c
Method	infix(INFIX *in, bool first)

```
....
597.                    sprintf(in->cur, " )");
```

Unchecked Return Value\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=846
Status	New

The infix method calls the sprintf function, at line 565 of PolarDB-for-PostgreSQL-2/_int_bool.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	PolarDB-for-PostgreSQL-2/_int_bool.c	PolarDB-for-PostgreSQL-2/_int_bool.c
Line	610	610
Object	sprintf	sprintf

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/_int_bool.c
Method	infix(INFIX *in, bool first)

```
....
610.                    sprintf(in->cur, "( ");
```

Unchecked Return Value\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=847
Status	New

The infix method calls the sprintf function, at line 565 of PolarDB-for-PostgreSQL-2/_int_bool.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	PolarDB-for-PostgreSQL-2/_int_bool.c	PolarDB-for-PostgreSQL-2/_int_bool.c
Line	627	627
Object	sprintf	sprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/_int_bool.c
Method	infix(INFIX *in, bool first)

```
....
627.                sprintf(in->cur, " %c %s", op, nrm.buf);
```

Unchecked Return Value\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=848
Status	New

The infix method calls the sprintf function, at line 565 of PolarDB-for-PostgreSQL-2/_int_bool.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	PolarDB-for-PostgreSQL-2/_int_bool.c	PolarDB-for-PostgreSQL-2/_int_bool.c
Line	634	634
Object	sprintf	sprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/_int_bool.c

Method	infix(INFIX *in, bool first)

```
....
634.                    sprintf(in->cur, " )");
```

Unchecked Return Value\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=849
Status	New

The ExecGrant_Largeobject method calls the snprintf function, at line 2749 of PolarDB-for-PostgreSQL-2/aclchk.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	PolarDB-for-PostgreSQL-2/aclchk.c	PolarDB-for-PostgreSQL-2/aclchk.c
Line	2832	2832
Object	snprintf	snprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/aclchk.c
Method	ExecGrant_Largeobject(InternalGrant *istmt)

```
....
2832.            snprintf(loname, sizeof(loname), "large object %u",
loid);
```

Unchecked Return Value\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=850
Status	New

The transformSetOperationTree method calls the snprintf function, at line 1842 of PolarDB-for-PostgreSQL-2/analyze.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	PolarDB-for-PostgreSQL-2/analyze.c	PolarDB-for-PostgreSQL-2/analyze.c
Line	1953	1953
Object	snprintf	snprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/analyze.c

Method	transformSetOperationTree(ParseState *pstate, SelectStmt *stmt,

```
....
1953.            snprintf(selectName, sizeof(selectName), "*SELECT*
%d",
```

Unchecked Return Value\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=851
Status	New

The apw_dump_now method calls the snprintf function, at line 574 of PolarDB-for-PostgreSQL-2/autoprewarm.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	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	637	637
Object	snprintf	snprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/autoprewarm.c
Method	apw_dump_now(bool is_bgworker, bool dump_unlogged)

```
....
637.        snprintf(transient_dump_file_path, MAXPGPATH, "%s.tmp",
AUTOPREWARM_FILE);
```

Unchecked Return Value\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=852
Status	New

The autovac_report_activity method calls the snprintf function, at line 3166 of PolarDB-for-PostgreSQL-2/autovacuum.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	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	3174	3174
Object	snprintf	snprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/autovacuum.c
Method	autovac_report_activity(autovac_table *tab)

```
....
3174.             snprintf(activity, MAX_AUTOVAC_ACTIV_LEN,
```

Unchecked Return Value\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=853
Status	New

The autovac_report_activity method calls the snprintf function, at line 3166 of PolarDB-for-PostgreSQL-2/autovacuum.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	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	3178	3178
Object	snprintf	snprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/autovacuum.c
Method	autovac_report_activity(autovac_table *tab)

```
....
3178.             snprintf(activity, MAX_AUTOVAC_ACTIV_LEN,
```

Unchecked Return Value\Path 12:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=854
Status	New

The autovac_report_activity method calls the snprintf function, at line 3166 of PolarDB-for-PostgreSQL-2/autovacuum.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	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	3186	3186
Object	snprintf	snprintf

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/autovacuum.c
Method	autovac_report_activity(autovac_table *tab)

```
....
3186.       snprintf(activity + len, MAX_AUTOVAC_ACTIV_LEN - len,
```

Unchecked Return Value\Path 13:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=855
Status	New

The autovac_report_workitem method calls the snprintf function, at line 3201 of PolarDB-for-PostgreSQL-2/autovacuum.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	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	3211	3211
Object	snprintf	snprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/autovacuum.c
Method	autovac_report_workitem(AutoVacuumWorkItem *workitem,

```
....
3211.                   snprintf(activity, MAX_AUTOVAC_ACTIV_LEN,
```

Unchecked Return Value\Path 14:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=856
Status	New

The autovac_report_workitem method calls the snprintf function, at line 3201 of PolarDB-for-PostgreSQL-2/autovacuum.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	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	3222	3222
Object	snprintf	snprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/autovacuum.c

Method	autovac_report_workitem(AutoVacuumWorkItem *workitem,

```
....
3222.             snprintf(blk, sizeof(blk), " %u", workitem-
>avw_blockNumber);
```

Unchecked Return Value\Path 15:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=857
Status	New

The autovac_report_workitem method calls the snprintf function, at line 3201 of PolarDB-for-PostgreSQL-2/autovacuum.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	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	3226	3226
Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method autovac_report_workitem(AutoVacuumWorkItem *workitem,

```
....
3226.       snprintf(activity + len, MAX_AUTOVAC_ACTIV_LEN - len,
```

Unchecked Return Value\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=858
Status	New

The SSLerrmessage method calls the snprintf function, at line 1044 of PolarDB-for-PostgreSQL-2/be-secure-openssl.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	PolarDB-for-PostgreSQL-2/be-secure-openssl.c	PolarDB-for-PostgreSQL-2/be-secure-openssl.c
Line	1054	1054
Object	snprintf	snprintf

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/be-secure-openssl.c
Method	SSLerrmessage(unsigned long ecode)

```
....
1054.       snprintf(errbuf, sizeof(errbuf), _("SSL error code %lu"),
ecode);
```

Unchecked Return Value\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=859
Status	New

The bt_page_print_tuples method calls the sprintf function, at line 256 of PolarDB-for-PostgreSQL-2/btreefuncs.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	PolarDB-for-PostgreSQL-2/btreefuncs.c	PolarDB-for-PostgreSQL-2/btreefuncs.c
Line	292	292
Object	sprintf	sprintf

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/btreefuncs.c
Method	bt_page_print_tuples(FuncCallContext *fctx, Page page, OffsetNumber offset)

```
....
292.                sprintf(dump, "%02x", *(ptr + off) & 0xff);
```

Unchecked Return Value\Path 18:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=860
Status	New

The copydir method calls the snprintf function, at line 37 of PolarDB-for-PostgreSQL-2/copydir.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	PolarDB-for-PostgreSQL-2/copydir.c	PolarDB-for-PostgreSQL-2/copydir.c
Line	62	62
Object	snprintf	snprintf

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/copydir.c
Method	copydir(char *fromdir, char *todir, bool recurse)

```
....
62.         snprintf(fromfile, sizeof(fromfile), "%s/%s", fromdir, xlde-
>d_name);
```

Unchecked Return Value\Path 19:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=861
Status	New

The copydir method calls the snprintf function, at line 37 of PolarDB-for-PostgreSQL-2/copydir.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	PolarDB-for-PostgreSQL-2/copydir.c	PolarDB-for-PostgreSQL-2/copydir.c
Line	63	63
Object	snprintf	snprintf

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/copydir.c
Method	copydir(char *fromdir, char *todir, bool recurse)

```
....
63.         snprintf(tofile, sizeof(tofile), "%s/%s", todir, xlde-
>d_name);
```

Unchecked Return Value\Path 20:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=862
Status	New

The copydir method calls the snprintf function, at line 37 of PolarDB-for-PostgreSQL-2/copydir.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	PolarDB-for-PostgreSQL-2/copydir.c	PolarDB-for-PostgreSQL-2/copydir.c
Line	98	98
Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copydir.c
Method copydir(char *fromdir, char *todir, bool recurse)

```
....
98.         snprintf(tofile, sizeof(tofile), "%s/%s", todir, xlde-
>d_name);
```

Unchecked Return Value\Path 21:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=863
Status	New

The db_dir_size method calls the snprintf function, at line 42 of PolarDB-for-PostgreSQL-2/dbsize.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	PolarDB-for-PostgreSQL-2/dbsize.c	PolarDB-for-PostgreSQL-2/dbsize.c
Line	66	66
Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbsize.c
Method db_dir_size(const char *path)

```
....
66.         snprintf(filename, sizeof(filename), "%s/%s",
polar_fullpath, direntry->d_name);
```

Unchecked Return Value\Path 22:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=864
Status	New

The calculate_database_size method calls the snprintf function, at line 88 of PolarDB-for-PostgreSQL-2/dbsize.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	PolarDB-for-PostgreSQL-2/dbsize.c	PolarDB-for-PostgreSQL-2/dbsize.c
Line	112	112
Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbsize.c
Method calculate_database_size(Oid dbOid)

```
....
112.        snprintf(pathname, sizeof(pathname), "base/%u", dbOid);
```

Unchecked Return Value\Path 23:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=865
Status	New

The calculate_database_size method calls the snprintf function, at line 88 of PolarDB-for-PostgreSQL-2/dbsize.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	PolarDB-for-PostgreSQL-2/dbsize.c	PolarDB-for-PostgreSQL-2/dbsize.c
Line	127	127
Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbsize.c
Method calculate_database_size(Oid dbOid)

```
....
127.              snprintf(pathname, sizeof(pathname),
"pg_tblspc/%s/%s/%u",
```

Unchecked Return Value\Path 24:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=866
Status	New

The calculate_tablespace_size method calls the snprintf function, at line 172 of PolarDB-for-PostgreSQL-2/dbsize.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	PolarDB-for-PostgreSQL-2/dbsize.c	PolarDB-for-PostgreSQL-2/dbsize.c
Line	197	197
Object	snprintf	snprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/dbsize.c
Method	calculate_tablespace_size(Oid tblspcOid)

```
....
197.              snprintf(tblspcPath, MAXPGPATH, "base");
```

Unchecked Return Value\Path 25:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=867
Status	New

The calculate_tablespace_size method calls the snprintf function, at line 172 of PolarDB-for-PostgreSQL-2/dbsize.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	PolarDB-for-PostgreSQL-2/dbsize.c	PolarDB-for-PostgreSQL-2/dbsize.c
Line	199	199
Object	snprintf	snprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/dbsize.c
Method	calculate_tablespace_size(Oid tblspcOid)

```
....
199.              snprintf(tblspcPath, MAXPGPATH, "global");
```

Unchecked Return Value\Path 26:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=868
Status	New

The calculate_tablespace_size method calls the snprintf function, at line 172 of PolarDB-for-PostgreSQL-2/dbsize.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	PolarDB-for-PostgreSQL-2/dbsize.c	PolarDB-for-PostgreSQL-2/dbsize.c
Line	201	201
Object	snprintf	snprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/dbsize.c
Method	calculate_tablespace_size(Oid tblspcOid)

```
....
201.                snprintf(tblspcPath, MAXPGPATH, "pg_tblspc/%u/%s",
tblspcOid,
```

Unchecked Return Value\Path 27:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=869
Status	New

The calculate_tablespace_size method calls the snprintf function, at line 172 of PolarDB-for-PostgreSQL-2/dbsize.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	PolarDB-for-PostgreSQL-2/dbsize.c	PolarDB-for-PostgreSQL-2/dbsize.c
Line	221	221
Object	snprintf	snprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/dbsize.c
Method	calculate_tablespace_size(Oid tblspcOid)

```
....
221.                snprintf(polar_tmp_path, sizeof(pathname), "%s/%s",
tblspcPath, direntry->d_name);
```

Unchecked Return Value\Path 28:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=870
Status	New

The calculate_tablespace_size method calls the snprintf function, at line 172 of PolarDB-for-PostgreSQL-2/dbsize.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	PolarDB-for-PostgreSQL-2/dbsize.c	PolarDB-for-PostgreSQL-2/dbsize.c
Line	222	222
Object	snprintf	snprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/dbsize.c
Method	calculate_tablespace_size(Oid tblspcOid)

```
....
222.                snprintf(pathname, sizeof(pathname), "%s/%s",
polar_full_path, direntry->d_name);
```

Unchecked Return Value\Path 29:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=871
Status	New

The calculate_relation_size method calls the snprintf function, at line 281 of PolarDB-for-PostgreSQL-2/dbsize.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	PolarDB-for-PostgreSQL-2/dbsize.c	PolarDB-for-PostgreSQL-2/dbsize.c
Line	297	297
Object	snprintf	snprintf

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/dbsize.c
Method	calculate_relation_size(RelFileNode *rfn, BackendId backend, ForkNumber forknum)

```
....
297.                    snprintf(pathname, MAXPGPATH, "%s",
```

Unchecked Return Value\Path 30:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=872
Status	New

The calculate_relation_size method calls the snprintf function, at line 281 of PolarDB-for-PostgreSQL-2/dbsize.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	PolarDB-for-PostgreSQL-2/dbsize.c	PolarDB-for-PostgreSQL-2/dbsize.c
Line	300	300

Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbsize.c
Method calculate_relation_size(RelFileNode *rfn, BackendId backend, ForkNumber forknum)

```
....
300.                    snprintf(pathname, MAXPGPATH, "%s.%u",
```

Unchecked Return Value\Path 31:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=873
Status New

The InitializeGUCOptionsFromEnvironment method calls the sprintf function, at line 8778 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8810	8810
Object	sprintf	sprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method InitializeGUCOptionsFromEnvironment(void)

```
....
8810.                   sprintf(limbuf, "%ld", new_limit);
```

Unchecked Return Value\Path 32:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=874
Status New

The SelectConfigFiles method calls the sprintf function, at line 8947 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8982	8982

Object	sprintf	sprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method SelectConfigFiles(const char *userDoption, const char *progname)

```
....
8982.             sprintf(fname, "%s/%s", configdir, CONFIG_FILENAME);
```

Unchecked Return Value\Path 33:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=875
Status New

The SelectConfigFiles method calls the sprintf function, at line 8947 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	9009	9009
Object	sprintf	sprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method SelectConfigFiles(const char *userDoption, const char *progname)

```
....
9009.             sprintf(fname, "%s/%s", configdir,
POLAR_DMA_FILENAME);
```

Unchecked Return Value\Path 34:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=876
Status New

The SelectConfigFiles method calls the sprintf function, at line 8947 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	9098	9098

Object	sprintf	sprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method SelectConfigFiles(const char *userDoption, const char *progname)

```
....
9098.             sprintf(fname, "%s/%s", configdir, HBA_FILENAME);
```

Unchecked Return Value\Path 35:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=877
Status	New

The SelectConfigFiles method calls the sprintf function, at line 8947 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	9121	9121
Object	sprintf	sprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method SelectConfigFiles(const char *userDoption, const char *progname)

```
....
9121.             sprintf(fname, "%s/%s", configdir, IDENT_FILENAME);
```

Unchecked Return Value\Path 36:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=878
Status	New

The GetConfigOption method calls the snprintf function, at line 10983 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	11012	11012

Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method GetConfigOption(const char *name, bool missing_ok, bool restrict_privileged)

```
....
11012.                  snprintf(buffer, sizeof(buffer), "%d",
```

Unchecked Return Value\Path 37:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=879
Status	New

The GetConfigOption method calls the snprintf function, at line 10983 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	11017	11017
Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method GetConfigOption(const char *name, bool missing_ok, bool restrict_privileged)

```
....
11017.                  snprintf(buffer, sizeof(buffer), "%g",
```

Unchecked Return Value\Path 38:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=880
Status	New

The GetConfigOptionResetString method calls the snprintf function, at line 11039 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	11062	11062

Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method GetConfigOptionResetString(const char *name)

```
....
11062.                  snprintf(buffer, sizeof(buffer), "%d",
```

Unchecked Return Value\Path 39:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=881
Status	New

The GetConfigOptionResetString method calls the snprintf function, at line 11039 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	11067	11067
Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method GetConfigOptionResetString(const char *name)

```
....
11067.                  snprintf(buffer, sizeof(buffer), "%g",
```

Unchecked Return Value\Path 40:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=882
Status	New

The AlterSystemSetConfigFile method calls the snprintf function, at line 11378 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	11485	11485

Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method AlterSystemSetConfigFile(AlterSystemStmt *altersysstmt)

```
....
11485.      snprintf(AutoConfFileName, sizeof(AutoConfFileName), "%s",
```

Unchecked Return Value\Path 41:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=883
Status New

The AlterSystemSetConfigFile method calls the snprintf function, at line 11378 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	11487	11487
Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method AlterSystemSetConfigFile(AlterSystemStmt *altersysstmt)

```
....
11487.      snprintf(AutoConfTmpFileName, sizeof(AutoConfTmpFileName),
"%s.%s",
```

Unchecked Return Value\Path 42:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=884
Status New

The polar_show_max_connections method calls the snprintf function, at line 12258 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	12267	12267

Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method polar_show_max_connections(void)

```
....
12267.      snprintf(nbuf, sizeof(nbuf), "%d", result);
```

Unchecked Return Value\Path 43:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=885
Status	New

The GetConfigOptionByNum method calls the snprintf function, at line 12409 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	12452	12452
Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method GetConfigOptionByNum(int varnum, const char **values, bool *noshow)

```
....
12452.                          snprintf(buffer, sizeof(buffer), "%dkB",
BLCKSZ / 1024);
```

Unchecked Return Value\Path 44:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=886
Status	New

The GetConfigOptionByNum method calls the snprintf function, at line 12409 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	12456	12456

Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method GetConfigOptionByNum(int varnum, const char **values, bool *noshow)

```
....
12456.                          snprintf(buffer, sizeof(buffer), "%dkB",
XLOG_BLCKSZ / 1024);
```

Unchecked Return Value\Path 45:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=887
Status New

The GetConfigOptionByNum method calls the snprintf function, at line 12409 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	12528	12528
Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method GetConfigOptionByNum(int varnum, const char **values, bool *noshow)

```
....
12528.                          snprintf(buffer, sizeof(buffer), "%d",
lconf->min);
```

Unchecked Return Value\Path 46:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=888
Status New

The GetConfigOptionByNum method calls the snprintf function, at line 12409 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c

Line	12532	12532
Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method GetConfigOptionByNum(int varnum, const char **values, bool *noshow)

```
....
12532.                          snprintf(buffer, sizeof(buffer), "%d",
lconf->max);
```

Unchecked Return Value\Path 47:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=889
Status New

The GetConfigOptionByNum method calls the snprintf function, at line 12409 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	12539	12539
Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method GetConfigOptionByNum(int varnum, const char **values, bool *noshow)

```
....
12539.                          snprintf(buffer, sizeof(buffer), "%d",
lconf->boot_val);
```

Unchecked Return Value\Path 48:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=890
Status New

The GetConfigOptionByNum method calls the snprintf function, at line 12409 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	12543	12543
Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method GetConfigOptionByNum(int varnum, const char **values, bool *noshow)

```
....
12543.                          snprintf(buffer, sizeof(buffer), "%d",
lconf->reset_val);
```

Unchecked Return Value\Path 49:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=891
Status New

The GetConfigOptionByNum method calls the snprintf function, at line 12409 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	12553	12553
Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method GetConfigOptionByNum(int varnum, const char **values, bool *noshow)

```
....
12553.                          snprintf(buffer, sizeof(buffer), "%g",
lconf->min);
```

Unchecked Return Value\Path 50:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=892
Status New

The GetConfigOptionByNum method calls the snprintf function, at line 12409 of PolarDB-for-PostgreSQL-2/guc.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	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	12557	12557
Object	snprintf	snprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method GetConfigOptionByNum(int varnum, const char **values, bool *noshow)

```
....
12557.                          snprintf(buffer, sizeof(buffer), "%g",
lconf->max);
```

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=1020028&projectid=20023&pathid=792
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/misc.c	PolarDB-for-PostgreSQL-2/misc.c
Line	985	985
Object	fgets	fgets

Code Snippet
File Name PolarDB-for-PostgreSQL-2/misc.c
Method pg_current_logfile(PG_FUNCTION_ARGS)

```
....
985.        while (fgets(lbuffer, sizeof(lbuffer), fd) != NULL)
```

Improper Resource Access Authorization\Path 2:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=793

Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	324	324
Object	fscanf	fscanf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
324.        if (fscanf(file, "<<%d>>\n", &num_elements) != 1)
```

Improper Resource Access Authorization\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=794
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	347	347
Object	fscanf	fscanf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
347.              if (fscanf(file, "%u,%u,%u,%u,%u\n",
&blkinfo[i].database,
```

Improper Resource Access Authorization\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=795
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c

Line	13212	13212
Object	fgetc	fgetc

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method read_string_with_null(FILE *fp)

```
....
13212.            if ((ch = fgetc(fp)) == EOF)
```

Improper Resource Access Authorization\Path 5:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=796
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/misc.c	PolarDB-for-PostgreSQL-2/misc.c
Line	985	985
Object	lbuffer	lbuffer

Code Snippet
File Name PolarDB-for-PostgreSQL-2/misc.c
Method pg_current_logfile(PG_FUNCTION_ARGS)

```
....
985.        while (fgets(lbuffer, sizeof(lbuffer), fd) != NULL)
```

Improper Resource Access Authorization\Path 6:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=797
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13280	13280
Object	Address	Address

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method read_nondefault_variables(void)

```
....
13280.            if (fread(&varsourceline, 1, sizeof(varsourceline),
fp) != sizeof(varsourceline))
```

Improper Resource Access Authorization\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=798
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13282	13282
Object	Address	Address

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method read_nondefault_variables(void)

```
....
13282.            if (fread(&varsource, 1, sizeof(varsource), fp) !=
sizeof(varsource))
```

Improper Resource Access Authorization\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=799
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13284	13284
Object	Address	Address

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method read_nondefault_variables(void)

```
....
13284.            if (fread(&varscontext, 1, sizeof(varscontext), fp) !=
sizeof(varscontext))
```

Improper Resource Access Authorization\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=800
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	324	324
Object	Address	Address

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
324.        if (fscanf(file, "<<%d>>\n", &num_elements) != 1)
```

Improper Resource Access Authorization\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=801
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	347	347
Object	Address	Address

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
347.                if (fscanf(file, "%u,%u,%u,%u,%u\n",
&blkinfo[i].database,
```

Improper Resource Access Authorization\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=802
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	348	348
Object	Address	Address

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
348.                                &blkinfo[i].tablespace,
&blkinfo[i].filenode,
```

Improper Resource Access Authorization\Path 12:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=803
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	348	348
Object	Address	Address

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
348.                                &blkinfo[i].tablespace,
&blkinfo[i].filenode,
```

Improper Resource Access Authorization\Path 13:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=804
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c

Line	349	349
Object	Address	Address

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
349.                                &forknum, &blkinfo[i].blocknum) != 5)
```

Improper Resource Access Authorization\Path 14:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=805
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	349	349
Object	Address	Address

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_load_buffers(void)

```
....
349.                                &forknum, &blkinfo[i].blocknum) != 5)
```

Improper Resource Access Authorization\Path 15:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=806
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/copydir.c	PolarDB-for-PostgreSQL-2/copydir.c
Line	190	190
Object	buffer	buffer

Code Snippet
File Name PolarDB-for-PostgreSQL-2/copydir.c
Method copy_file(char *fromfile, char *tofile)

```
....
190.                nbytes = read(srcfd, buffer, COPY_BUF_SIZE);
```

Improper Resource Access Authorization\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=807
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/origin.c	PolarDB-for-PostgreSQL-2/origin.c
Line	722	722
Object	Address	Address

Code Snippet
File Name PolarDB-for-PostgreSQL-2/origin.c
Method StartupReplicationOrigin(void)

```
....
722.        readBytes = read(fd, &magic, sizeof(magic));
```

Improper Resource Access Authorization\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=808
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/origin.c	PolarDB-for-PostgreSQL-2/origin.c
Line	741	741
Object	Address	Address

Code Snippet
File Name PolarDB-for-PostgreSQL-2/origin.c
Method StartupReplicationOrigin(void)

```
....
741.                readBytes = read(fd, &disk_state, sizeof(disk_state));
```

Improper Resource Access Authorization\Path 18:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=809
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/misc.c	PolarDB-for-PostgreSQL-2/misc.c
Line	543	543
Object	targetpath	targetpath

Code Snippet
File Name PolarDB-for-PostgreSQL-2/misc.c
Method pg_tablespace_location(PG_FUNCTION_ARGS)

```
....
543.        rllen = readlink(sourcepath, targetpath,
sizeof(targetpath));
```

Improper Resource Access Authorization\Path 19:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=810
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autoprewarm.c	PolarDB-for-PostgreSQL-2/autoprewarm.c
Line	645	645
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_dump_now(bool is_bgworker, bool dump_unlogged)

```
....
645.        ret = fprintf(file, "<<%d>>\n", num_blocks);
```

Improper Resource Access Authorization\Path 20:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=811
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-	PolarDB-for-PostgreSQL-

	2/autoprewarm.c	2/autoprewarm.c
Line	663	663
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autoprewarm.c
Method apw_dump_now(bool is_bgworker, bool dump_unlogged)

```
....
663.                ret = fprintf(file, "%u,%u,%u,%u,%u\n",
```

Improper Resource Access Authorization\Path 21:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=812
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13090	13090
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_one_nondefault_variable(FILE *fp, struct config_generic *gconf)

```
....
13090.      fprintf(fp, "%s", gconf->name);
```

Improper Resource Access Authorization\Path 22:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=813
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13100	13100
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_one_nondefault_variable(FILE *fp, struct config_generic *gconf)

```
....
13100.                                  fprintf(fp, "true");
```

Improper Resource Access Authorization\Path 23:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=814
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13102	13102
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_one_nondefault_variable(FILE *fp, struct config_generic *gconf)

```
....
13102.                                  fprintf(fp, "false");
```

Improper Resource Access Authorization\Path 24:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=815
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13110	13110
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_one_nondefault_variable(FILE *fp, struct config_generic *gconf)

```
....
13110.                          fprintf(fp, "%d", *conf->variable);
```

Improper Resource Access Authorization\Path 25:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=816
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13118	13118
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_one_nondefault_variable(FILE *fp, struct config_generic *gconf)

```
....
13118.                          fprintf(fp, "%.17g", *conf->variable);
```

Improper Resource Access Authorization\Path 26:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=817
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13126	13126
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_one_nondefault_variable(FILE *fp, struct config_generic *gconf)

```
....
13126.                          fprintf(fp, "%s", *conf->variable);
```

Improper Resource Access Authorization\Path 27:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=818
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13134	13134

Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_one_nondefault_variable(FILE *fp, struct config_generic *gconf)

```
....
13134.                          fprintf(fp, "%s",
```

Improper Resource Access Authorization\Path 28:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=819
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13143	13143
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_one_nondefault_variable(FILE *fp, struct config_generic *gconf)

```
....
13143.              fprintf(fp, "%s", gconf->sourcefile);
```

Improper Resource Access Authorization\Path 29:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=820
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuumdb.c	PolarDB-for-PostgreSQL-2/vacuumdb.c
Line	203	203
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuumdb.c
Method main(int argc, char *argv[])

```
....
203.                                    fprintf(stderr, _("%s: number of
parallel jobs must be at least 1\n"),
```

Improper Resource Access Authorization\Path 30:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=821
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuumdb.c	PolarDB-for-PostgreSQL-2/vacuumdb.c
Line	215	215
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuumdb.c
Method main(int argc, char *argv[])

```
....
215.                            fprintf(stderr, _("Try \"%s --help\" for
more information.\n"), progname);
```

Improper Resource Access Authorization\Path 31:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=822
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuumdb.c	PolarDB-for-PostgreSQL-2/vacuumdb.c
Line	232	232
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuumdb.c
Method main(int argc, char *argv[])

```
....
232.                fprintf(stderr, _("%s: too many command-line arguments
(first is \"%s\")\n"),
```

Improper Resource Access Authorization\Path 32:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=823
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuumdb.c	PolarDB-for-PostgreSQL-2/vacuumdb.c
Line	234	234
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuumdb.c
Method main(int argc, char *argv[])

```
....
234.                fprintf(stderr, _("Try \"%s --help\" for more
information.\n"), progname);
```

Improper Resource Access Authorization\Path 33:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=824
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuumdb.c	PolarDB-for-PostgreSQL-2/vacuumdb.c
Line	242	242
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuumdb.c
Method main(int argc, char *argv[])

```
....
242.                    fprintf(stderr, _("%s: cannot use the \"%s\"
option when performing only analyze\n"),
```

Improper Resource Access Authorization\Path 34:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=825
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuumdb.c	PolarDB-for-PostgreSQL-2/vacuumdb.c
Line	248	248
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuumdb.c
Method main(int argc, char *argv[])

```
....
248.                     fprintf(stderr, _("%s: cannot use the \"%s\"
option when performing only analyze\n"),
```

Improper Resource Access Authorization\Path 35:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=826
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuumdb.c	PolarDB-for-PostgreSQL-2/vacuumdb.c
Line	265	265
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuumdb.c
Method main(int argc, char *argv[])

```
....
265.                     fprintf(stderr, _("%s: cannot vacuum all
databases and a specific one at the same time\n"),
```

Improper Resource Access Authorization\Path 36:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=827
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuumdb.c	PolarDB-for-PostgreSQL-2/vacuumdb.c
Line	271	271
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuumdb.c
Method main(int argc, char *argv[])

```
....
271.                    fprintf(stderr, _("%s: cannot vacuum specific
table(s) in all databases\n"),
```

Improper Resource Access Authorization\Path 37:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=828
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuumdb.c	PolarDB-for-PostgreSQL-2/vacuumdb.c
Line	451	451
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuumdb.c
Method vacuum_one_database(const char *dbname, vacuumingOptions *vacopts,

```
....
451.                            fprintf(stderr,
```

Improper Resource Access Authorization\Path 38:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=829
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuumdb.c	PolarDB-for-PostgreSQL-2/vacuumdb.c
Line	738	738
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuumdb.c
Method run_vacuum_command(PGconn *conn, const char *sql, bool echo,

```
....
738.                    fprintf(stderr,
```

Improper Resource Access Authorization\Path 39:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=830
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuumdb.c	PolarDB-for-PostgreSQL-2/vacuumdb.c
Line	742	742
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuumdb.c
Method run_vacuum_command(PGconn *conn, const char *sql, bool echo,

```
....
742.                    fprintf(stderr, _("%s: vacuuming of database
\"%s\" failed: %s"),
```

Improper Resource Access Authorization\Path 40:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=831
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuumdb.c	PolarDB-for-PostgreSQL-2/vacuumdb.c
Line	877	877
Object	fprintf	fprintf

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuumdb.c
Method ProcessQueryResult(PGconn *conn, PGresult *result, const char *progname)

```
....
877.                fprintf(stderr, _("%s: vacuuming of database \"%s\"
failed: %s"),
```

Improper Resource Access Authorization\Path 41:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=832
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13091	13091
Object	fputc	fputc

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_one_nondefault_variable(FILE *fp, struct config_generic *gconf)

```
....
13091.      fputc(0, fp);
```

Improper Resource Access Authorization\Path 42:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=833
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13140	13140
Object	fputc	fputc

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_one_nondefault_variable(FILE *fp, struct config_generic *gconf)

```
....
13140.      fputc(0, fp);
```

Improper Resource Access Authorization\Path 43:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=834
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13144	13144
Object	fputc	fputc

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_one_nondefault_variable(FILE *fp, struct config_generic *gconf)

```
....
13144.      fputc(0, fp);
```

Improper Resource Access Authorization\Path 44:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=835
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13146	13146
Object	fwrite	fwrite

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_one_nondefault_variable(FILE *fp, struct config_generic *gconf)

```
....
13146.      fwrite(&gconf->sourceline, 1, sizeof(gconf->sourceline),
fp);
```

Improper Resource Access Authorization\Path 45:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=836
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13147	13147
Object	fwrite	fwrite

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_one_nondefault_variable(FILE *fp, struct config_generic *gconf)

```
....
13147.      fwrite(&gconf->source, 1, sizeof(gconf->source), fp);
```

Improper Resource Access Authorization\Path 46:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=837
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	13148	13148
Object	fwrite	fwrite

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method write_one_nondefault_variable(FILE *fp, struct config_generic *gconf)

```
....
13148.      fwrite(&gconf->scontext, 1, sizeof(gconf->scontext), fp);
```

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=1020028&projectid=20023&pathid=979
Status	New

The variable declared in null at PolarDB-for-PostgreSQL-2/autovacuum.c in line 1153 is not initialized when it is used by adw_entry at PolarDB-for-PostgreSQL-2/autovacuum.c in line 1153.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	1231	1310
Object	null	adw_entry

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method do_start_worker(void)

```
....
1231.       avdb = NULL;
....
1310.                   tmp->adw_entry->last_autovac_time < avdb-
>adw_entry->last_autovac_time)
```

NULL Pointer Dereference\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=980
Status	New

The variable declared in null at PolarDB-for-PostgreSQL-2/execProcnode.c in line 147 is not initialized when it is used by result at PolarDB-for-PostgreSQL-2/execProcnode.c in line 147.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execProcnode.c	PolarDB-for-PostgreSQL-2/execProcnode.c
Line	412	416
Object	null	result

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execProcnode.c
Method ExecInitNode(Plan *node, EState *estate, int eflags)

```
....
412.                    result = NULL;          /* keep compiler quiet
*/
....
416.        ExecSetExecProcNode(result, result->ExecProcNode);
```

NULL Pointer Dereference\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=981
Status	New

The variable declared in null at PolarDB-for-PostgreSQL-2/functions.c in line 477 is not initialized when it is used by stmt at PolarDB-for-PostgreSQL-2/functions.c in line 477.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/functions.c	PolarDB-for-PostgreSQL-2/functions.c
Line	482	590
Object	null	stmt

Code Snippet
File Name PolarDB-for-PostgreSQL-2/functions.c
Method init_execution_state(List *queryTree_list,

```
....
482.        execution_state *lasttages = NULL;
....
590.                    lasttages->stmt->commandType == CMD_SELECT &&
```

NULL Pointer Dereference\Path 4:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=982
Status New

The variable declared in null at PolarDB-for-PostgreSQL-2/functions.c in line 477 is not initialized when it is used by stmt at PolarDB-for-PostgreSQL-2/functions.c in line 477.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/functions.c	PolarDB-for-PostgreSQL-2/functions.c
Line	482	591
Object	null	stmt

Code Snippet
File Name PolarDB-for-PostgreSQL-2/functions.c
Method init_execution_state(List *queryTree_list,

```
....
482.        execution_state *lasttages = NULL;
....
591.                    !lasttages->stmt->hasModifyingCTE)
```

NULL Pointer Dereference\Path 5:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=983
Status New

The variable declared in null at PolarDB-for-PostgreSQL-2/functions.c in line 1602 is not initialized when it is used by parse at PolarDB-for-PostgreSQL-2/functions.c in line 1602.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/functions.c	PolarDB-for-PostgreSQL-2/functions.c
Line	1633	1926
Object	null	parse

Code Snippet
File Name PolarDB-for-PostgreSQL-2/functions.c
Method check_sql_fn_retval(Oid func_id, Oid rettype, List *queryTreeList,

```
....
1633.       parse = NULL;
....
1926.                           if (parse->setOperations)
```

NULL Pointer Dereference\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=984
Status	New

The variable declared in null at PolarDB-for-PostgreSQL-2/functions.c in line 1602 is not initialized when it is used by parse at PolarDB-for-PostgreSQL-2/functions.c in line 1602.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/functions.c	PolarDB-for-PostgreSQL-2/functions.c
Line	1633	1864
Object	null	parse

Code Snippet
File Name PolarDB-for-PostgreSQL-2/functions.c
Method check_sql_fn_retval(Oid func_id, Oid rettype, List *queryTreeList,

```
....
1633.       parse = NULL;
....
1864.                                   if (parse->setOperations)
```

NULL Pointer Dereference\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=985
Status	New

The variable declared in null at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194 is not initialized when it is used by lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ltree_io.c	PolarDB-for-PostgreSQL-2/ltree_io.c
Line	203	434
Object	null	lptr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ltree_io.c
Method lquery_in(PG_FUNCTION_ARGS)

```
....
203.         nodeitem   *lptr = NULL;
....
434.                     lptr->wlen++;
```

NULL Pointer Dereference\Path 8:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=986
Status New

The variable declared in null at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194 is not initialized when it is used by lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ltree_io.c	PolarDB-for-PostgreSQL-2/ltree_io.c
Line	203	282
Object	null	lptr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ltree_io.c
Method lquery_in(PG_FUNCTION_ARGS)

```
....
203.         nodeitem   *lptr = NULL;
....
282.                             lptr->flag |= LVAR_INCASE;
```

NULL Pointer Dereference\Path 9:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=987
Status New

The variable declared in null at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194 is not initialized when it is used by lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ltree_io.c	PolarDB-for-PostgreSQL-2/ltree_io.c
Line	203	289
Object	null	lptr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ltree_io.c
Method lquery_in(PG_FUNCTION_ARGS)

```
....
203.        nodeitem   *lptr = NULL;
....
289.                           lptr->flag |= LVAR_ANYEND;
```

NULL Pointer Dereference\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=988
Status	New

The variable declared in null at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194 is not initialized when it is used by lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ltree_io.c	PolarDB-for-PostgreSQL-2/ltree_io.c
Line	203	296
Object	null	lptr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ltree_io.c
Method lquery_in(PG_FUNCTION_ARGS)

```
....
203.        nodeitem   *lptr = NULL;
....
296.                           lptr->flag |= LVAR_SUBLEXEME;
```

NULL Pointer Dereference\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=989
Status	New

The variable declared in null at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194 is not initialized when it is used by lptr at PolarDB-for-PostgreSQL-2/ltree_io.c in line 194.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ltree_io.c	PolarDB-for-PostgreSQL-2/ltree_io.c
Line	203	334
Object	null	lptr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ltree_io.c
Method lquery_in(PG_FUNCTION_ARGS)

```
....
203.        nodeitem   *lptr = NULL;
....
334.                           if (lptr->flag)
```

NULL Pointer Dereference\Path 12:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=990
Status New

The variable declared in null at PolarDB-for-PostgreSQL-2/origin.c in line 858 is not initialized when it is used by replication_state at PolarDB-for-PostgreSQL-2/origin.c in line 858.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/origin.c	PolarDB-for-PostgreSQL-2/origin.c
Line	863	970
Object	null	replication_state

Code Snippet
File Name PolarDB-for-PostgreSQL-2/origin.c
Method replorigin_advance(RepOriginId node,

```
....
863.        ReplicationState *replication_state = NULL;
....
970.        LWLockRelease(&replication_state->lock);
```

NULL Pointer Dereference\Path 13:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=991
Status New

The variable declared in null at PolarDB-for-PostgreSQL-2/origin.c in line 858 is not initialized when it is used by replication_state at PolarDB-for-PostgreSQL-2/origin.c in line 858.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/origin.c	PolarDB-for-PostgreSQL-2/origin.c
Line	863	939
Object	null	replication_state

Code Snippet
File Name PolarDB-for-PostgreSQL-2/origin.c
Method replorigin_advance(RepOriginId node,

```
....
863.        ReplicationState *replication_state = NULL;
....
939.        Assert(replication_state->roident != InvalidRepOriginId);
```

NULL Pointer Dereference\Path 14:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=992
Status	New

The variable declared in null at PolarDB-for-PostgreSQL-2/origin.c in line 858 is not initialized when it is used by free_state at PolarDB-for-PostgreSQL-2/origin.c in line 858.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/origin.c	PolarDB-for-PostgreSQL-2/origin.c
Line	864	932
Object	null	free_state

Code Snippet
File Name PolarDB-for-PostgreSQL-2/origin.c
Method replorigin_advance(RepOriginId node,

```
....
864.        ReplicationState *free_state = NULL;
....
932.              LWLockAcquire(&free_state->lock, LW_EXCLUSIVE);
```

NULL Pointer Dereference\Path 15:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=993
Status	New

The variable declared in null at PolarDB-for-PostgreSQL-2/trigger.c in line 4213 is not initialized when it is used by LocTriggerData at PolarDB-for-PostgreSQL-2/trigger.c in line 4213.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/trigger.c	PolarDB-for-PostgreSQL-2/trigger.c
Line	4290	4287
Object	null	LocTriggerData

Code Snippet
File Name PolarDB-for-PostgreSQL-2/trigger.c
Method AfterTriggerExecute(AfterTriggerEvent event,

```
....
4290.                           ExecMaterializeSlot(trig_tuple_slot2) :
NULL;
....
4287.                    LocTriggerData.tg_newtuple =
```

NULL Pointer Dereference\Path 16:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=994
Status	New

The variable declared in null at PolarDB-for-PostgreSQL-2/trigger.c in line 4976 is not initialized when it is used by events at PolarDB-for-PostgreSQL-2/trigger.c in line 3982.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/trigger.c	PolarDB-for-PostgreSQL-2/trigger.c
Line	5000	4050
Object	null	events

Code Snippet
File Name PolarDB-for-PostgreSQL-2/trigger.c
Method AfterTriggerFireDeferred(void)

```
....
5000.       while (afterTriggerMarkEvents(events, NULL, false))
```

File Name PolarDB-for-PostgreSQL-2/trigger.c
Method afterTriggerAddEvent(AfterTriggerEventList *events,

```
....
4050.             if (events->head == NULL)
```

NULL Pointer Dereference\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=995
Status	New

The variable declared in null at PolarDB-for-PostgreSQL-2/trigger.c in line 5352 is not initialized when it is used by events at PolarDB-for-PostgreSQL-2/trigger.c in line 3982.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/trigger.c	PolarDB-for-PostgreSQL-2/trigger.c
Line	5615	4050
Object	null	events

Code Snippet
File Name PolarDB-for-PostgreSQL-2/trigger.c
Method AfterTriggerSetState(ConstraintsSetStmt *stmt)

```
....
5615.             while (afterTriggerMarkEvents(events, NULL, true))
```

File Name PolarDB-for-PostgreSQL-2/trigger.c

Method afterTriggerAddEvent(AfterTriggerEventList *events,

```
....
4050.             if (events->head == NULL)
```

NULL Pointer Dereference\Path 18:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=996
Status New

The variable declared in null at PolarDB-for-PostgreSQL-2/vacuum.c in line 1318 is not initialized when it is used by relation at PolarDB-for-PostgreSQL-2/vacuum.c in line 1318.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuum.c	PolarDB-for-PostgreSQL-2/vacuum.c
Line	1600	1436
Object	null	relation

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuum.c
Method vacuum_rel(Oid relid, RangeVar *relation, int options, VacuumParams *params)

```
....
1600.             vacuum_rel(toast_relid, NULL, options, params);
....
1436.                                                 relation-
>relname)));
```

NULL Pointer Dereference\Path 19:

Severity Low
Result State To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=997
Status	New

The variable declared in null at PolarDB-for-PostgreSQL-2/vacuum.c in line 1318 is not initialized when it is used by relation at PolarDB-for-PostgreSQL-2/vacuum.c in line 1318.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuum.c	PolarDB-for-PostgreSQL-2/vacuum.c
Line	1600	1441
Object	null	relation

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuum.c
Method vacuum_rel(Oid relid, RangeVar *relation, int options, VacuumParams *params)

```
....
1600.             vacuum_rel(toast_relid, NULL, options, params);
....
1441.                                                 relation->relname)));
```

NULL Pointer Dereference\Path 20:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=998
Status	New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/geo_ops.c in line 1389 is not initialized when it is used by path at PolarDB-for-PostgreSQL-2/geo_ops.c in line 1389.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/geo_ops.c	PolarDB-for-PostgreSQL-2/geo_ops.c
Line	1410	1410
Object	0	path

Code Snippet
File Name PolarDB-for-PostgreSQL-2/geo_ops.c
Method path_recv(PG_FUNCTION_ARGS)

```
....
1410.       path->closed = (closed ? 1 : 0);
```

NULL Pointer Dereference\Path 21:

Severity	Low
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=999
Status	New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/heap.c in line 2516 is not initialized when it is used by cooked at PolarDB-for-PostgreSQL-2/heap.c in line 2516.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/heap.c	PolarDB-for-PostgreSQL-2/heap.c
Line	2600	2600
Object	0	cooked

Code Snippet
File Name PolarDB-for-PostgreSQL-2/heap.c
Method AddRelationNewConstraints(Relation rel,

```
....
2600.             cooked->inhcount = is_local ? 0 : 1;
```

NULL Pointer Dereference\Path 22:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1000
Status	New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/heap.c in line 2516 is not initialized when it is used by cooked at PolarDB-for-PostgreSQL-2/heap.c in line 2516.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/heap.c	PolarDB-for-PostgreSQL-2/heap.c
Line	2731	2731
Object	0	cooked

Code Snippet
File Name PolarDB-for-PostgreSQL-2/heap.c
Method AddRelationNewConstraints(Relation rel,

```
....
2731.             cooked->inhcount = is_local ? 0 : 1;
```

NULL Pointer Dereference\Path 23:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1001

Status	New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/nodeSetOp.c in line 150 is not initialized when it is used by setopstate at PolarDB-for-PostgreSQL-2/nodeSetOp.c in line 150.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeSetOp.c	PolarDB-for-PostgreSQL-2/nodeSetOp.c
Line	176	174
Object	0	setopstate

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeSetOp.c
Method set_output_count(SetOpState *setopstate, SetOpStatePerGroup pergroup)

```
....
176.                            0 : (pergroup->numLeft - pergroup-
>numRight);
....
174.                      setopstate->numOutput =
```

NULL Pointer Dereference\Path 24:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1002
Status	New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/nodeSetOp.c in line 150 is not initialized when it is used by setopstate at PolarDB-for-PostgreSQL-2/nodeSetOp.c in line 227.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeSetOp.c	PolarDB-for-PostgreSQL-2/nodeSetOp.c
Line	176	251
Object	0	setopstate

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeSetOp.c
Method set_output_count(SetOpState *setopstate, SetOpStatePerGroup pergroup)

```
....
176.                            0 : (pergroup->numLeft - pergroup-
>numRight);
```

File Name PolarDB-for-PostgreSQL-2/nodeSetOp.c

Method setop_retrieve_direct(SetOpState *setopstate)

```
....
251.              if (setopstate->grp_firstTuple == NULL)
```

NULL Pointer Dereference\Path 25:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1003
Status	New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/nodeSetOp.c in line 150 is not initialized when it is used by setopstate at PolarDB-for-PostgreSQL-2/nodeSetOp.c in line 227.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeSetOp.c	PolarDB-for-PostgreSQL-2/nodeSetOp.c
Line	176	245
Object	0	setopstate

Code Snippet

File Name PolarDB-for-PostgreSQL-2/nodeSetOp.c

Method set_output_count(SetOpState *setopstate, SetOpStatePerGroup pergroup)

```
....
176.                            0 : (pergroup->numLeft - pergroup-
>numRight);
```

File Name PolarDB-for-PostgreSQL-2/nodeSetOp.c

Method setop_retrieve_direct(SetOpState *setopstate)

```
....
245.        while (!setopstate->setop_done)
```

NULL Pointer Dereference\Path 26:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1004
Status	New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/nodeSetOp.c in line 150 is not initialized when it is used by setopstate at PolarDB-for-PostgreSQL-2/nodeSetOp.c in line 427.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeSetOp.c	PolarDB-for-PostgreSQL-2/nodeSetOp.c
Line	176	440

Object	0	setopstate

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeSetOp.c
Method set_output_count(SetOpState *setopstate, SetOpStatePerGroup pergroup)

```
....
176.                            0 : (pergroup->numLeft - pergroup-
>numRight);
```

File Name PolarDB-for-PostgreSQL-2/nodeSetOp.c

Method setop_retrieve_hash_table(SetOpState *setopstate)

```
....
440.        while (!setopstate->setop_done)
```

NULL Pointer Dereference\Path 27:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1005
Status	New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/nodeSubplan.c in line 1046 is not initialized when it is used by prm at PolarDB-for-PostgreSQL-2/nodeSubplan.c in line 1046.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeSubplan.c	PolarDB-for-PostgreSQL-2/nodeSubplan.c
Line	1227	1227
Object	0	prm

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeSubplan.c
Method ExecSetParamPlan(SubPlanState *node, ExprContext *econtext)

```
....
1227.                           prm->value = (Datum) 0;
```

NULL Pointer Dereference\Path 28:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1006
Status	New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/rangetypes.c in line 162 is not initialized when it is used by lower at PolarDB-for-PostgreSQL-2/rangetypes.c in line 162.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/rangetypes.c	PolarDB-for-PostgreSQL-2/rangetypes.c
Line	208	208
Object	0	lower

Code Snippet
File Name PolarDB-for-PostgreSQL-2/rangetypes.c
Method range_recv(PG_FUNCTION_ARGS)

```
....
208.                lower.val = (Datum) 0;
```

NULL Pointer Dereference\Path 29:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1007
Status New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/rangetypes.c in line 162 is not initialized when it is used by upper at PolarDB-for-PostgreSQL-2/rangetypes.c in line 162.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/rangetypes.c	PolarDB-for-PostgreSQL-2/rangetypes.c
Line	226	226
Object	0	upper

Code Snippet
File Name PolarDB-for-PostgreSQL-2/rangetypes.c
Method range_recv(PG_FUNCTION_ARGS)

```
....
226.                upper.val = (Datum) 0;
```

NULL Pointer Dereference\Path 30:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1008
Status New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/rangetypes.c in line 360 is not initialized when it is used by lower at PolarDB-for-PostgreSQL-2/rangetypes.c in line 360.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/rangetypes.c	PolarDB-for-PostgreSQL-2/rangetypes.c
Line	372	372
Object	0	lower

Code Snippet
File Name PolarDB-for-PostgreSQL-2/rangetypes.c
Method range_constructor2(PG_FUNCTION_ARGS)

```
....
372.        lower.val = PG_ARGISNULL(0) ? (Datum) 0 : arg1;
```

NULL Pointer Dereference\Path 31:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1009
Status New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/rangetypes.c in line 360 is not initialized when it is used by upper at PolarDB-for-PostgreSQL-2/rangetypes.c in line 360.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/rangetypes.c	PolarDB-for-PostgreSQL-2/rangetypes.c
Line	377	377
Object	0	upper

Code Snippet
File Name PolarDB-for-PostgreSQL-2/rangetypes.c
Method range_constructor2(PG_FUNCTION_ARGS)

```
....
377.        upper.val = PG_ARGISNULL(1) ? (Datum) 0 : arg2;
```

NULL Pointer Dereference\Path 32:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1010
Status New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/rangetypes.c in line 389 is not initialized when it is used by lower at PolarDB-for-PostgreSQL-2/rangetypes.c in line 389.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/rangetypes.c	PolarDB-for-PostgreSQL-2/rangetypes.c

Line	409	409
Object	0	lower

Code Snippet
File Name PolarDB-for-PostgreSQL-2/rangetypes.c
Method range_constructor3(PG_FUNCTION_ARGS)

```
....
409.        lower.val = PG_ARGISNULL(0) ? (Datum) 0 : arg1;
```

NULL Pointer Dereference\Path 33:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1011
Status	New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/rangetypes.c in line 389 is not initialized when it is used by upper at PolarDB-for-PostgreSQL-2/rangetypes.c in line 389.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/rangetypes.c	PolarDB-for-PostgreSQL-2/rangetypes.c
Line	414	414
Object	0	upper

Code Snippet
File Name PolarDB-for-PostgreSQL-2/rangetypes.c
Method range_constructor3(PG_FUNCTION_ARGS)

```
....
414.        upper.val = PG_ARGISNULL(1) ? (Datum) 0 : arg2;
```

NULL Pointer Dereference\Path 34:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1012
Status	New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/rangetypes.c in line 1943 is not initialized when it is used by lower at PolarDB-for-PostgreSQL-2/rangetypes.c in line 1943.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/rangetypes.c	PolarDB-for-PostgreSQL-2/rangetypes.c
Line	1948	1948
Object	0	lower

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/rangetypes.c
Method	make_empty_range(TypeCacheEntry *typcache)

```
....
1948.       lower.val = (Datum) 0;
```

NULL Pointer Dereference\Path 35:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1013
Status	New

The variable declared in 0 at PolarDB-for-PostgreSQL-2/rangetypes.c in line 1943 is not initialized when it is used by upper at PolarDB-for-PostgreSQL-2/rangetypes.c in line 1943.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/rangetypes.c	PolarDB-for-PostgreSQL-2/rangetypes.c
Line	1953	1953
Object	0	upper

Code Snippet	
File Name	PolarDB-for-PostgreSQL-2/rangetypes.c
Method	make_empty_range(TypeCacheEntry *typcache)

```
....
1953.       upper.val = (Datum) 0;
```

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=1020028&projectid=20023&pathid=1015
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c

Line	373	373
Object	ac	ac

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method avlauncher_forkexec(void)

```
....
373.        av[ac] = NULL;
```

Unchecked Array Index\Path 2:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1016
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/autovacuum.c	PolarDB-for-PostgreSQL-2/autovacuum.c
Line	1481	1481
Object	ac	ac

Code Snippet
File Name PolarDB-for-PostgreSQL-2/autovacuum.c
Method avworker_forkexec(void)

```
....
1481.       av[ac] = NULL;
```

Unchecked Array Index\Path 3:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1017
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/btreefuncs.c	PolarDB-for-PostgreSQL-2/btreefuncs.c
Line	287	287
Object	j	j

Code Snippet
File Name PolarDB-for-PostgreSQL-2/btreefuncs.c
Method bt_page_print_tuples(FuncCallContext *fctx, Page page, OffsetNumber offset)

```
....
287.        values[j] = dump;
```

Unchecked Array Index\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1018
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c
Line	2259	2259
Object	argno	argno

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExecInitFunc(ExprEvalStep *scratch, Expr *node, List *args, Oid funcid,

```
....
2259.                   fcinfo->arg[argno] = con->constvalue;
```

Unchecked Array Index\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1019
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c
Line	2260	2260
Object	argno	argno

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExecInitFunc(ExprEvalStep *scratch, Expr *node, List *args, Oid funcid,

```
....
2260.                   fcinfo->argnull[argno] = con->constisnull;
```

Unchecked Array Index\Path 6:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1020
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c
Line	2529	2529
Object	i	i

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExecInitArrayRef(ExprEvalStep *scratch, ArrayRef *aref,

```
....
2529.                   arefstate->upperprovided[i] = false;
```

Unchecked Array Index\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1021
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c
Line	2534	2534
Object	i	i

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExecInitArrayRef(ExprEvalStep *scratch, ArrayRef *aref,

```
....
2534.              arefstate->upperprovided[i] = true;
```

Unchecked Array Index\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1022
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c
Line	2562	2562

Object	i	i

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExecInitArrayRef(ExprEvalStep *scratch, ArrayRef *aref,

```
....
2562.                   arefstate->lowerprovided[i] = false;
```

Unchecked Array Index\Path 9:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1023
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/execExpr.c	PolarDB-for-PostgreSQL-2/execExpr.c
Line	2567	2567
Object	i	i

Code Snippet
File Name PolarDB-for-PostgreSQL-2/execExpr.c
Method ExecInitArrayRef(ExprEvalStep *scratch, ArrayRef *aref,

```
....
2567.              arefstate->lowerprovided[i] = true;
```

Unchecked Array Index\Path 10:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1024
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/funcapi.c	PolarDB-for-PostgreSQL-2/funcapi.c
Line	1104	1104
Object	numinargs	numinargs

Code Snippet
File Name PolarDB-for-PostgreSQL-2/funcapi.c
Method get_func_input_arg_names(Datum proargnames, Datum proargmodes,

```
....
1104.                           inargnames[numinargs] = pname;
```

Unchecked Array Index\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1025
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/funcapi.c	PolarDB-for-PostgreSQL-2/funcapi.c
Line	1106	1106
Object	numinargs	numinargs

Code Snippet
File Name PolarDB-for-PostgreSQL-2/funcapi.c
Method get_func_input_arg_names(Datum proargnames, Datum proargmodes,

```
....
1106.                           inargnames[numinargs] = NULL;
```

Unchecked Array Index\Path 12:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1026
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/indexcmds.c	PolarDB-for-PostgreSQL-2/indexcmds.c
Line	2114	2114
Object	ndx	ndx

Code Snippet
File Name PolarDB-for-PostgreSQL-2/indexcmds.c
Method makeObjectName(const char *name1, const char *name2, const char *label)

```
....
2114.              name[ndx] = '\0';
```

Unchecked Array Index\Path 13:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1027
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeIndexscan.c	PolarDB-for-PostgreSQL-2/nodeIndexscan.c
Line	410	410
Object	i	i

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeIndexscan.c
Method EvalOrderByExpressions(IndexScanState *node, ExprContext *econtext)

```
....
410.                node->iss_OrderByValues[i] = ExecEvalExpr(orderby,
```

Unchecked Array Index\Path 14:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1028
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeProjectSet.c	PolarDB-for-PostgreSQL-2/nodeProjectSet.c
Line	283	283
Object	off	off

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeProjectSet.c
Method ExecInitProjectSet(ProjectSet *node, EState *estate, int eflags)

```
....
283.                    state->elems[off] = (Node *)
```

Unchecked Array Index\Path 15:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1029
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-	PolarDB-for-PostgreSQL-

	2/nodeProjectSet.c	2/nodeProjectSet.c
Line	290	290
Object	off	off

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeProjectSet.c
Method ExecInitProjectSet(ProjectSet *node, EState *estate, int eflags)

```
....
290.                    state->elems[off] = (Node *) ExecInitExpr(expr,
&state->ps);
```

Unchecked Array Index\Path 16:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1030
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/plperl.c	PolarDB-for-PostgreSQL-2/plperl.c
Line	500	500
Object	OP_REQUIRE	OP_REQUIRE

Code Snippet
File Name PolarDB-for-PostgreSQL-2/plperl.c
Method set_interp_require(bool trusted)

```
....
500.              PL_ppaddr[OP_REQUIRE] = pp_require_safe;
```

Unchecked Array Index\Path 17:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1031
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/plperl.c	PolarDB-for-PostgreSQL-2/plperl.c
Line	501	501
Object	OP_DOFILE	OP_DOFILE

Code Snippet
File Name PolarDB-for-PostgreSQL-2/plperl.c

Method	set_interp_require(bool trusted)

```
....
501.              PL_ppaddr[OP_DOFILE] = pp_require_safe;
```

Unchecked Array Index\Path 18:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1032
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/plperl.c	PolarDB-for-PostgreSQL-2/plperl.c
Line	505	505
Object	OP_REQUIRE	OP_REQUIRE

Code Snippet
File Name PolarDB-for-PostgreSQL-2/plperl.c
Method set_interp_require(bool trusted)

```
....
505.              PL_ppaddr[OP_REQUIRE] = pp_require_orig;
```

Unchecked Array Index\Path 19:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1033
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/plperl.c	PolarDB-for-PostgreSQL-2/plperl.c
Line	506	506
Object	OP_DOFILE	OP_DOFILE

Code Snippet
File Name PolarDB-for-PostgreSQL-2/plperl.c
Method set_interp_require(bool trusted)

```
....
506.              PL_ppaddr[OP_DOFILE] = pp_require_orig;
```

Unchecked Array Index\Path 20:

Severity	Low
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1034
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/plperl.c	PolarDB-for-PostgreSQL-2/plperl.c
Line	837	837
Object	OP_REQUIRE	OP_REQUIRE

Code Snippet
File Name PolarDB-for-PostgreSQL-2/plperl.c
Method plperl_init_interp(void)

```
....
837.                    PL_ppaddr[OP_REQUIRE] = pp_require_orig;
```

Unchecked Array Index\Path 21:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1035
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/plperl.c	PolarDB-for-PostgreSQL-2/plperl.c
Line	838	838
Object	OP_DOFILE	OP_DOFILE

Code Snippet
File Name PolarDB-for-PostgreSQL-2/plperl.c
Method plperl_init_interp(void)

```
....
838.                    PL_ppaddr[OP_DOFILE] = pp_require_orig;
```

Unchecked Array Index\Path 22:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1036
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/plperl.c	PolarDB-for-PostgreSQL-2/plperl.c

Line	973	973
Object	OP_REQUIRE	OP_REQUIRE

Code Snippet
File Name PolarDB-for-PostgreSQL-2/plperl.c
Method plperl_trusted_init(void)

```
....
973.        PL_ppaddr[OP_REQUIRE] = pp_require_orig;
```

Unchecked Array Index\Path 23:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1037
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/plperl.c	PolarDB-for-PostgreSQL-2/plperl.c
Line	974	974
Object	OP_DOFILE	OP_DOFILE

Code Snippet
File Name PolarDB-for-PostgreSQL-2/plperl.c
Method plperl_trusted_init(void)

```
....
974.        PL_ppaddr[OP_DOFILE] = pp_require_orig;
```

Unchecked Array Index\Path 24:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1038
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/plperl.c	PolarDB-for-PostgreSQL-2/plperl.c
Line	1000	1000
Object	OP_REQUIRE	OP_REQUIRE

Code Snippet
File Name PolarDB-for-PostgreSQL-2/plperl.c
Method plperl_trusted_init(void)

```
....
1000.       PL_ppaddr[OP_REQUIRE] = pp_require_safe;
```

Unchecked Array Index\Path 25:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1039
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/plperl.c	PolarDB-for-PostgreSQL-2/plperl.c
Line	1001	1001
Object	OP_DOFILE	OP_DOFILE

Code Snippet
File Name PolarDB-for-PostgreSQL-2/plperl.c
Method plperl_trusted_init(void)

```
....
1001.       PL_ppaddr[OP_DOFILE] = pp_require_safe;
```

Unchecked Array Index\Path 26:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1040
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	1875	1875
Object	plength	plength

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres.c
Method exec_bind_message(StringInfo input_message)

```
....
1875.                       pbuf.data[plength] = '\0';
```

Unchecked Array Index\Path 27:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1041
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	1949	1949
Object	plength	plength

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres.c
Method exec_bind_message(StringInfo input_message)

```
....
1949.                           pbuf.data[plength] = csave;
```

Unchecked Array Index\Path 28:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1042
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/px_disp_query.c	PolarDB-for-PostgreSQL-2/px_disp_query.c
Line	324	324
Object	nthbyte	nthbyte

Code Snippet
File Name PolarDB-for-PostgreSQL-2/px_disp_query.c
Method mark_bit(char *bits, int nth)

```
....
324.        bits[nthbyte] |= nthbit;
```

Unchecked Array Index\Path 29:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1043
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/ri_triggers.c	PolarDB-for-PostgreSQL-2/ri_triggers.c

Line	1167	1167
Object	j	j

Code Snippet
File Name PolarDB-for-PostgreSQL-2/ri_triggers.c
Method RI_FKey_cascade_upd(PG_FUNCTION_ARGS)

```
....
1167.                                   queryoids[j] = pk_type;
```

Unchecked Array Index\Path 30:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1044
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/tsearchcmds.c	PolarDB-for-PostgreSQL-2/tsearchcmds.c
Line	1257	1257
Object	i	i

Code Snippet
File Name PolarDB-for-PostgreSQL-2/tsearchcmds.c
Method getTokenTypes(Oid prsId, List *tokennames)

```
....
1257.                           res[i] = list[j].lexid;
```

Unchecked Array Index\Path 31:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1045
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/tsvector_op.c	PolarDB-for-PostgreSQL-2/tsvector_op.c
Line	2369	2369
Object	values	values

Code Snippet
File Name PolarDB-for-PostgreSQL-2/tsvector_op.c
Method ts_process_call(FuncCallContext *funcctx)

```
....
2369.             (values[0])[entry->lenlexeme] = '\0';
```

Unchecked Array Index\Path 32:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1046
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/tsvector_op.c	PolarDB-for-PostgreSQL-2/tsvector_op.c
Line	2369	2369
Object	lenlexeme	lenlexeme

Code Snippet
File Name PolarDB-for-PostgreSQL-2/tsvector_op.c
Method ts_process_call(FuncCallContext *funcctx)

```
....
2369.             (values[0])[entry->lenlexeme] = '\0';
```

Unchecked Array Index\Path 33:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=1047
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/xact.c	PolarDB-for-PostgreSQL-2/xact.c
Line	1582	1582
Object	nChildXids	nChildXids

Code Snippet
File Name PolarDB-for-PostgreSQL-2/xact.c
Method AtSubCommit_childXids(void)

```
....
1582.       s->parent->childXids[s->parent->nChildXids] = s-
>transactionId;
```

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=1020028&projectid=20023&pathid=957
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/xact.c	PolarDB-for-PostgreSQL-2/xact.c
Line	220	562
Object	TransactionState	sizeof

Code Snippet

File Name PolarDB-for-PostgreSQL-2/xact.c

Method typedef TransactionStateData *TransactionState;

```
....
220.  typedef TransactionStateData *TransactionState;
```

File Name PolarDB-for-PostgreSQL-2/xact.c

Method AssignTransactionId(TransactionState s)

```
....
562.              parents = palloc(sizeof(TransactionState) * s-
>nestingLevel);
```

Use of Sizeof On a Pointer Type\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=958
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/_ltree_op.c	PolarDB-for-PostgreSQL-2/_ltree_op.c
Line	309	309
Object	sizeof	sizeof

Code Snippet

File Name PolarDB-for-PostgreSQL-2/_ltree_op.c

Method _lca(PG_FUNCTION_ARGS)

```
....
309.        a = (ltree **) palloc(sizeof(ltree *) * num);
```

Use of Sizeof On a Pointer Type\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=959
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/analyze.c	PolarDB-for-PostgreSQL-2/analyze.c
Line	1426	1426
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/analyze.c
Method transformValuesClause(ParseState *pstate, SelectStmt *stmt)

```
....
1426.                    colexprs = (List **) palloc0(sublist_length *
sizeof(List *));
```

Use of Sizeof On a Pointer Type\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=960
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/event_trigger.c	PolarDB-for-PostgreSQL-2/event_trigger.c
Line	698	698
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/event_trigger.c
Method filter_event_trigger(const char **tag, EventTriggerCacheItem *item)

```
....
698.                                                    item->ntags,
sizeof(char *),
```

Use of Sizeof On a Pointer Type\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=200

	23&pathid=961
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/funcapi.c	PolarDB-for-PostgreSQL-2/funcapi.c
Line	960	960
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/funcapi.c
Method get_func_arg_info(HeapTuple procTup,

```
....
960.                *p_argnames = (char **) palloc(sizeof(char *) *
numargs);
```

Use of Sizeof On a Pointer Type\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=962
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/funcapi.c	PolarDB-for-PostgreSQL-2/funcapi.c
Line	1092	1092
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/funcapi.c
Method get_func_input_arg_names(Datum proargnames, Datum proargmodes,

```
....
1092.       inargnames = (char **) palloc(numargs * sizeof(char *));
```

Use of Sizeof On a Pointer Type\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=963
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/funcapi.c	PolarDB-for-PostgreSQL-2/funcapi.c
Line	1340	1340

Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/funcapi.c
Method build_function_result_tupdesc_d(char prokind,

```
....
1340.       outargnames = (char **) palloc(numargs * sizeof(char *));
```

Use of Sizeof On a Pointer Type\Path 8:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=964
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8473	8473
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method build_guc_variables(void)

```
....
8473.             guc_malloc(FATAL, size_vars * sizeof(struct
config_generic *));
```

Use of Sizeof On a Pointer Type\Path 9:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=965
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8528	8528
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method build_guc_variables(void)

```
....
8528.                sizeof(struct config_generic *), guc_var_compare);
```

Use of Sizeof On a Pointer Type\Path 10:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=966
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8550	8550
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method add_guc_variable(struct config_generic *var, int elevel)

```
....
8550.                          guc_malloc(elevel, size_vars *
sizeof(struct config_generic *));
```

Use of Sizeof On a Pointer Type\Path 11:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=967
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8555	8555
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method add_guc_variable(struct config_generic *var, int elevel)

```
....
8555.                          guc_realloc(elevel, guc_variables,
size_vars * sizeof(struct config_generic *));
```

Use of Sizeof On a Pointer Type\Path 12:

Severity	Low

Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=968
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8571	8571
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method add_guc_variable(struct config_generic *var, int elevel)

```
....
8571.                sizeof(struct config_generic *), guc_var_compare);
```

Use of Sizeof On a Pointer Type\Path 13:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=969
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8581	8581
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method add_placeholder_variable(const char *name, int elevel)

```
....
8581.        size_t            sz = sizeof(struct config_string) +
sizeof(char *);
```

Use of Sizeof On a Pointer Type\Path 14:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=970
Status	New

	Source	Destination

File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8642	8642
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method find_option(const char *name, bool create_placeholders, int elevel)

```
....
8642.
sizeof(struct config_generic *),
```

Use of Sizeof On a Pointer Type\Path 15:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=971
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	11877	11877
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method define_custom_variable(struct config_generic *variable)

```
....
11877.
sizeof(struct config_generic *),
```

Use of Sizeof On a Pointer Type\Path 16:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=972
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	15047	15047
Object	sizeof	sizeof

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/guc.c
Method	polar_check_is_forbidden_funcs(List *funcname)

```
....
15047.                                  sizeof(char *), pg_qsort_strcmp))
```

Use of Sizeof On a Pointer Type\Path 17:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=973
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	15110	15110
Object	sizeof	sizeof

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/guc.c
Method	assign_polar_forbidden_functions(const char *newval, void *extra)

```
....
15110.      polar_forbidden_functions = guc_malloc(FATAL,
num_forbidden_functions * sizeof(char *));
```

Use of Sizeof On a Pointer Type\Path 18:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=974
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	15123	15123
Object	sizeof	sizeof

Code Snippet

File Name	PolarDB-for-PostgreSQL-2/guc.c
Method	assign_polar_forbidden_functions(const char *newval, void *extra)

```
....
15123.      qsort(polar_forbidden_functions, num_forbidden_functions,
sizeof(char *), pg_qsort_strcmp);
```

Use of Sizeof On a Pointer Type\Path 19:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=975
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/nodeProjectSet.c	PolarDB-for-PostgreSQL-2/nodeProjectSet.c
Line	264	264
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/nodeProjectSet.c
Method ExecInitProjectSet(ProjectSet *node, EState *estate, int eflags)

```
....
264.              palloc(sizeof(Node *) * state->nelems);
```

Use of Sizeof On a Pointer Type\Path 20:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=976
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/trigger.c	PolarDB-for-PostgreSQL-2/trigger.c
Line	1999	1999
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/trigger.c
Method RelationBuildTriggers(Relation relation)

```
....
1999.                    build->tgargs = (char **) palloc(build->tgnargs
* sizeof(char *));
```

Use of Sizeof On a Pointer Type\Path 21:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=977
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/trigger.c	PolarDB-for-PostgreSQL-2/trigger.c
Line	2176	2176
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/trigger.c
Method CopyTriggerDesc(TriggerDesc *trigdesc)

```
....
2176.                   newargs = (char **) palloc(trigger->tgnargs *
sizeof(char *));
```

Use of Sizeof On a Pointer Type\Path 22:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=978
Status New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/tsvector_op.c	PolarDB-for-PostgreSQL-2/tsvector_op.c
Line	2269	2269
Object	sizeof	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/tsvector_op.c
Method ts_setup_firstcall(FunctionCallInfo fcinfo, FuncCallContext *funcctx,

```
....
2269.       stat->stack = palloc0(sizeof(StatEntry *) * (stat->maxdepth
+ 1));
```

Potential Precision Problem

Query Path:
CPP\Cx\CPP Buffer Overflow\Potential Precision Problem Version:0

Categories

NIST SP 800-53: SI-10 Information Input Validation (P1)
OWASP Top 10 2017: A1-Injection

Description

Potential Precision Problem\Path 1:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=200

	23&pathid=268
Status	New

The size of the buffer used by infix in " %c %s", at line 565 of PolarDB-for-PostgreSQL-2/_int_bool.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that infix passes to " %c %s", at line 565 of PolarDB-for-PostgreSQL-2/_int_bool.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/_int_bool.c	PolarDB-for-PostgreSQL-2/_int_bool.c
Line	627	627
Object	" %c %s"	" %c %s"

Code Snippet
File Name PolarDB-for-PostgreSQL-2/_int_bool.c
Method infix(INFIX *in, bool first)

```
....
627.                sprintf(in->cur, " %c %s", op, nrm.buf);
```

Potential Precision Problem\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=269
Status	New

The size of the buffer used by SelectConfigFiles in "%s/%s", at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that SelectConfigFiles passes to "%s/%s", at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	8982	8982
Object	"%s/%s"	"%s/%s"

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method SelectConfigFiles(const char *userDoption, const char *progname)

```
....
8982.               sprintf(fname, "%s/%s", configdir, CONFIG_FILENAME);
```

Potential Precision Problem\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=270

Status	New

The size of the buffer used by SelectConfigFiles in "%s/%s", at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that SelectConfigFiles passes to "%s/%s", at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	9009	9009
Object	"%s/%s"	"%s/%s"

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method SelectConfigFiles(const char *userDoption, const char *progname)

```
....
9009.             sprintf(fname, "%s/%s", configdir,
POLAR_DMA_FILENAME);
```

Potential Precision Problem\Path 4:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=271
Status	New

The size of the buffer used by SelectConfigFiles in "%s/%s", at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that SelectConfigFiles passes to "%s/%s", at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	9098	9098
Object	"%s/%s"	"%s/%s"

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method SelectConfigFiles(const char *userDoption, const char *progname)

```
....
9098.             sprintf(fname, "%s/%s", configdir, HBA_FILENAME);
```

Potential Precision Problem\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=272

Status	New

The size of the buffer used by SelectConfigFiles in "%s/%s", at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that SelectConfigFiles passes to "%s/%s", at line 8947 of PolarDB-for-PostgreSQL-2/guc.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	9121	9121
Object	"%s/%s"	"%s/%s"

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method SelectConfigFiles(const char *userDoption, const char *progname)

```
....
9121.             sprintf(fname, "%s/%s", configdir, IDENT_FILENAME);
```

Potential Precision Problem\Path 6:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=273
Status	New

The size of the buffer used by assign_pgstat_temp_directory in "%s", at line 14812 of PolarDB-for-PostgreSQL-2/guc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that assign_pgstat_temp_directory passes to "%s", at line 14812 of PolarDB-for-PostgreSQL-2/guc.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	14821	14821
Object	"%s"	"%s"

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method assign_pgstat_temp_directory(const char *newval, void *extra)

```
....
14821.      sprintf(dname, "%s", newval);
```

Potential Precision Problem\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=274
Status	New

The size of the buffer used by assign_pgstat_temp_directory in "%s/global.tmp", at line 14812 of PolarDB-for-PostgreSQL-2/guc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that assign_pgstat_temp_directory passes to "%s/global.tmp", at line 14812 of PolarDB-for-PostgreSQL-2/guc.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	14825	14825
Object	"%s/global.tmp"	"%s/global.tmp"

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method assign_pgstat_temp_directory(const char *newval, void *extra)

```
....
14825.      sprintf(tname, "%s/global.tmp", newval);
```

Potential Precision Problem\Path 8:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=275
Status	New

The size of the buffer used by assign_pgstat_temp_directory in "%s/global.stat", at line 14812 of PolarDB-for-PostgreSQL-2/guc.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that assign_pgstat_temp_directory passes to "%s/global.stat", at line 14812 of PolarDB-for-PostgreSQL-2/guc.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	14827	14827
Object	"%s/global.stat"	"%s/global.stat"

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method assign_pgstat_temp_directory(const char *newval, void *extra)

```
....
14827.      sprintf(fname, "%s/global.stat", newval);
```

Potential Precision Problem\Path 9:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=276
Status	New

The size of the buffer used by plperl_create_sub in "%s__%u", at line 2087 of PolarDB-for-PostgreSQL-2/plperl.c, is not properly verified before writing data to the buffer. This can enable a buffer overflow attack, using the source buffer that plperl_create_sub passes to "%s__%u", at line 2087 of PolarDB-for-PostgreSQL-2/plperl.c, to overwrite the target buffer.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/plperl.c	PolarDB-for-PostgreSQL-2/plperl.c
Line	2096	2096
Object	"%s__%u"	"%s__%u"

Code Snippet
File Name PolarDB-for-PostgreSQL-2/plperl.c
Method plperl_create_sub(plperl_proc_desc *prodesc, const char *s, Oid fn_oid)

```
....
2096.       sprintf(subname, "%s__%u", prodesc->proname, fn_oid);
```

Potential Off by One Error in Loops

Query Path:
CPP\Cx\CPP Heuristic\Potential Off by One Error in Loops Version:1

Categories

PCI DSS v3.2: PCI DSS (3.2) - 6.5.1 - Injection flaws - particularly SQL injection
NIST SP 800-53: SI-16 Memory Protection (P1)
OWASP Top 10 2017: A1-Injection

Description
Potential Off by One Error in Loops\Path 1:

Severity Low
Result State To Verify
Online Results http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=50
Status New

The buffer allocated by <= in PolarDB-for-PostgreSQL-2/dbsize.c at line 351 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/dbsize.c	PolarDB-for-PostgreSQL-2/dbsize.c
Line	362	362
Object	<=	<=

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbsize.c
Method calculate_toast_table_size(Oid toastrelid)

```
....
362.        for (forkNum = 0; forkNum <= MAX_FORKNUM; forkNum++)
```

Potential Off by One Error in Loops\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=51
Status	New

The buffer allocated by <= in PolarDB-for-PostgreSQL-2/dbsize.c at line 351 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/dbsize.c	PolarDB-for-PostgreSQL-2/dbsize.c
Line	376	376
Object	<=	<=

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbsize.c
Method calculate_toast_table_size(Oid toastrelid)

```
....
376.                for (forkNum = 0; forkNum <= MAX_FORKNUM; forkNum++)
```

Potential Off by One Error in Loops\Path 3:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=52
Status	New

The buffer allocated by <= in PolarDB-for-PostgreSQL-2/dbsize.c at line 397 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/dbsize.c	PolarDB-for-PostgreSQL-2/dbsize.c
Line	405	405
Object	<=	<=

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbsize.c
Method calculate_table_size(Relation rel)

```
....
405.        for (forkNum = 0; forkNum <= MAX_FORKNUM; forkNum++)
```

Potential Off by One Error in Loops\Path 4:

Severity	Low
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=53
Status	New

The buffer allocated by <= in PolarDB-for-PostgreSQL-2/dbsize.c at line 424 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/dbsize.c	PolarDB-for-PostgreSQL-2/dbsize.c
Line	444	444
Object	<=	<=

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbsize.c
Method calculate_indexes_size(Relation rel)

```
....
444.                    for (forkNum = 0; forkNum <= MAX_FORKNUM;
forkNum++)
```

Potential Off by One Error in Loops\Path 5:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=54
Status	New

The buffer allocated by <= in PolarDB-for-PostgreSQL-2/guc.c at line 14328 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	14360	14360
Object	<=	<=

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method check_wal_consistency_checking(char **newval, void **extra, GucSource source)

```
....
14360.                  for (rmid = 0; rmid <= RM_MAX_ID; rmid++)
```

Potential Off by One Error in Loops\Path 6:

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

	BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=55
Status	New

The buffer allocated by <= in PolarDB-for-PostgreSQL-2/guc.c at line 14328 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/guc.c	PolarDB-for-PostgreSQL-2/guc.c
Line	14371	14371
Object	<=	<=

Code Snippet
File Name PolarDB-for-PostgreSQL-2/guc.c
Method check_wal_consistency_checking(char **newval, void **extra, GucSource source)

```
....
14371.                  for (rmid = 0; rmid <= RM_MAX_ID; rmid++)
```

Potential Off by One Error in Loops\Path 7:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=56
Status	New

The buffer allocated by <= in PolarDB-for-PostgreSQL-2/postgres.c at line 3917 does not correctly account for the actual size of the value, resulting in an incorrect allocation that is off by one.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	4387	4387
Object	<=	<=

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres.c
Method PostgresMain(int argc, char *argv[],

```
....
4387.                   for (rmid = 0; rmid <= RM_MAX_ID; rmid++)
```

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=1020028&projectid=20023&pathid=838
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/dbcommands.c	PolarDB-for-PostgreSQL-2/dbcommands.c
Line	665	665
Object	mkdir	mkdir

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbcommands.c
Method createdb(ParseState *pstate, const CreatedbStmt *stmt)

```
....
665.                            if (mkdir(dstpath, S_IRWXU) != 0)
```

Incorrect Permission Assignment For Critical Resources\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=839
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/dbcommands.c	PolarDB-for-PostgreSQL-2/dbcommands.c
Line	1325	1325
Object	mkdir	mkdir

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbcommands.c
Method movedb(const char *dbname, const char *tblspcname)

```
....
1325.                    if (mkdir(dst_dbpath, S_IRWXU) != 0)
```

Incorrect Permission Assignment For Critical Resources\Path 3:

Severity	Low
Result State	To Verify

Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=840
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/dbcommands.c	PolarDB-for-PostgreSQL-2/dbcommands.c
Line	2258	2258
Object	mkdir	mkdir

Code Snippet
File Name PolarDB-for-PostgreSQL-2/dbcommands.c
Method dbase_redo(XLogReaderState *record)

```
....
2258.                           if (mkdir(dst_path, S_IRWXU) != 0)
```

Inconsistent Implementations

Query Path:
CPP\Cx\CPP Low Visibility\Inconsistent Implementations Version:0

Description

Inconsistent Implementations\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=43
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	3697	3697
Object	getopt	getopt

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres.c
Method process_postgres_switches(int argc, char *argv[], GucContext ctx,

```
....
3697.       while ((flag = getopt(argc, argv,
"B:bc:C:D:d:EeFf:h:ijk:lN:nOo:Pp:r:S:sTt:v:W:-:")) != -1)
```

Inconsistent Implementations\Path 2:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=44

Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/vacuumdb.c	PolarDB-for-PostgreSQL-2/vacuumdb.c
Line	147	147
Object	getopt_long	getopt_long

Code Snippet
File Name PolarDB-for-PostgreSQL-2/vacuumdb.c
Method main(int argc, char *argv[])

```
....
147.         while ((c = getopt_long(argc, argv,
"h:p:U:wWeqd:zZFat:fvj:", long_options, &optindex)) != -1)
```

Arithmenic Operation On Boolean

Query Path:
CPP\Cx\CPP Low Visibility\Arithmenic Operation On Boolean Version:1

Categories

FISMA 2014: Audit And Accountability
NIST SP 800-53: SC-5 Denial of Service Protection (P1)

Description
Arithmenic Operation On Boolean\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=283
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/index.c	PolarDB-for-PostgreSQL-2/index.c
Line	4499	4499
Object	BinaryExpr	BinaryExpr

Code Snippet
File Name PolarDB-for-PostgreSQL-2/index.c
Method polar_px_validate_index_heapscan(Relation heapRelation,

```
....
4499.                                                   nulls + 1,
```

Use of Insufficiently Random Values

Query Path:
CPP\Cx\CPP Low Visibility\Use of Insufficiently Random Values Version:0

Categories

FISMA 2014: Media Protection
NIST SP 800-53: SC-28 Protection of Information at Rest (P1)
OWASP Top 10 2017: A3-Sensitive Data Exposure

Description

Use of Insufficiently Random Values\Path 1:

Severity	Low
Result State	To Verify
Online Results	http://WIN-BA8RD5TJ8IG/CxWebClient/ViewerMain.aspx?scanid=1020028&projectid=20023&pathid=842
Status	New

Method polar_hold_shared_storage at line 95 of PolarDB-for-PostgreSQL-2/polar_io_fencing.c uses a weak method random to produce random values. These values might be used for secret values, personal identifiers or cryptographic input, allowing an attacker to guess the value.

	Source	Destination
File	PolarDB-for-PostgreSQL-2/polar_io_fencing.c	PolarDB-for-PostgreSQL-2/polar_io_fencing.c
Line	107	107
Object	random	random

Code Snippet
File Name PolarDB-for-PostgreSQL-2/polar_io_fencing.c
Method polar_hold_shared_storage(bool force_hold)

```
....
107.                seed = random();
```

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=1020028&projectid=20023&pathid=1014
Status	New

	Source	Destination
File	PolarDB-for-PostgreSQL-2/heap.c	PolarDB-for-PostgreSQL-2/heap.c
Line	3465	3465
Object	nulls	sizeof

Code Snippet
File Name PolarDB-for-PostgreSQL-2/heap.c
Method StorePartitionKey(Relation rel,

```
....
3465.       MemSet(nulls, false, sizeof(nulls));
```

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=1020028&projectid=20023&pathid=1048
Status	New

The polar_save_stack_info method in PolarDB-for-PostgreSQL-2/postgres.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	PolarDB-for-PostgreSQL-2/postgres.c	PolarDB-for-PostgreSQL-2/postgres.c
Line	5299	5299
Object	open	open

Code Snippet
File Name PolarDB-for-PostgreSQL-2/postgres.c
Method polar_save_stack_info(void)

```
....
5299.       fd = open(polar_core_file, O_CREAT | O_WRONLY | PG_BINARY,
pg_file_create_mode);
```

Buffer Overflow LongString

Risk

What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

Cause

How does it happen

Buffer Overflows can manifest in numerous different variations. In it's most basic form, the attack controls a buffer, which is then copied to a smaller buffer without size verification. Because the attacker's source buffer is larger than the program's target buffer, the attacker's data overwrites whatever is next on the stack, allowing the attacker to control program structures.

Alternatively, the vulnerability could be the result of improper bounds checking; exposing internal memory addresses outside of their valid scope; allowing the attacker to control the size of the target buffer; or various other forms.

General Recommendations

How to avoid it

- Always perform proper bounds checking before copying buffers or strings.
- Prefer to use safer functions and structures, e.g. safe string classes over `char*`, `strncpy` over `strcpy`, and so on.
- Consistently apply tests for the size of buffers.
- Do not return variable addresses outside the scope of their variables.

Source Code Examples

CPP

Overflowing Buffers

```
const int BUFFER_SIZE = 10;
char buffer[BUFFER_SIZE];

void copyStringToBuffer(char* inputString)
{
    strcpy(buffer, inputString);
}
```

Checked Buffers

```
const int BUFFER_SIZE = 10;
const int MAX_INPUT_SIZE = 256;
char buffer[BUFFER_SIZE];

void copyStringToBuffer(char* inputString)
{
    if (strnlen(inputString, MAX_INPUT_SIZE) < sizeof(buffer))
    {
        strncpy(buffer, inputString, sizeof(buffer));
    }
}
```

Buffer Overflow StrcpyStrcat

Risk

What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

Cause

How does it happen

Buffer Overflows can manifest in numerous different variations. In it's most basic form, the attack controls a buffer, which is then copied to a smaller buffer without size verification. Because the attacker's source buffer is larger than the program's target buffer, the attacker's data overwrites whatever is next on the stack, allowing the attacker to control program structures.
Alternatively, the vulnerability could be the result of improper bounds checking; exposing internal memory addresses outside of their valid scope; allowing the attacker to control the size of the target buffer; or various other forms.

General Recommendations

How to avoid it

- Always perform proper bounds checking before copying buffers or strings.
- Prefer to use safer functions and structures, e.g. safe string classes over `char*`, `strncpy` over `strcpy`, and so on.
- Consistently apply tests for the size of buffers.
- Do not return variable addresses outside the scope of their variables.

Source Code Examples

Buffer Overflow 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;
}
```

Off by One Error in Methods

Risk

What might happen

An off by one error may result in overwriting or over-reading of unintended memory; in most cases, this can result in unexpected behavior and even application crashes. In other cases, where allocation can be controlled by an attacker, a combination of variable assignment and an off by one error can result in execution of malicious code.

Cause

How does it happen

Often when designating variables to memory, a calculation error may occur when determining size or length that is off by one.

For example in loops, when allocating an array of size 2, its cells are counted as 0,1 - therefore, if a For loop iterator on the array is incorrectly set with the start condition i=0 and the continuation condition i<=2, three cells will be accessed instead of 2, and an attempt will be made to write or read cell [2], which was not originally allocated, resulting in potential corruption of memory outside the bounds of the originally assigned array.

Another example occurs when a null-byte terminated string, in the form of a character array, is copied without its terminating null-byte. Without the null-byte, the string representation is unterminated, resulting in certain functions to over-read memory as they expect the missing null terminator.

General Recommendations

How to avoid it

- Always ensure that a given iteration boundary is correct:
  - With array iterations, consider that arrays begin with cell 0 and end with cell n-1, for a size n array.
  - With character arrays and null-byte terminated string representations, consider that the null byte is required and should not be overwritten or ignored; ensure functions in use are not vulnerable to off-by-one, specifically for instances where null-bytes are automatically appended after the buffer, instead of in place of its last character.
- Where possible, use safe functions that manage memory and are not prone to off-by-one errors.

Source Code Examples

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);
```

Missing Precision

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

Char Overflow

Risk

What might happen

Assigning large data types into smaller data types, without proper checks and explicit casting, will lead to undefined behavior and unintentional effects, such as data corruption (e.g. value wraparound, wherein maximum values become minimum values); system crashes; infinite loops; logic errors, such as bypassing of security mechanisms; or even buffer overflows leading to arbitrary code execution.

Cause

How does it happen

This flaw can occur when implicitly casting numerical data types of a larger size, into a variable with a data type of a smaller size. This forces the program to discard some bits of information from the number. Depending on how the numerical data types are stored in memory, this is often the bits with the highest value, causing substantial corruption of the stored number. Alternatively, the sign bit of a signed integer could be lost, completely reversing the intention of the number.

General Recommendations

How to avoid it

- Avoid casting larger data types to smaller types.
- Prefer promoting the target variable to a large enough data type.
- If downcasting is necessary, always check that values are valid and in range of the target type, before casting

---

Source Code Examples

CPP

Unsafe Downsize Casting

```
int unsafe_addition(short op1, int op2) {

    // op2 gets forced from int into a short
    short total = op1 + op2;

    return total;
}
```

Safer Use of Proper Data Types

```
int safe_addition(short op1, int op2) {

    // total variable is of type int, the largest type that is needed
    int total = 0;

    // check if total will overflow available integer size
    if (INT_MAX - abs(op2) > op1)
```

```
    {
        total = op1 + op2;
    }
    else
    {
        // instead of overflow, saturate (but this is not always a good thing)
        total = INT_MAX
    }

    return total;
}
```

Integer Overflow

Risk

What might happen

Assigning large data types into smaller data types, without proper checks and explicit casting, will lead to undefined behavior and unintentional effects, such as data corruption (e.g. value wraparound, wherein maximum values become minimum values); system crashes; infinite loops; logic errors, such as bypassing of security mechanisms; or even buffer overflows leading to arbitrary code execution.

Cause

How does it happen

This flaw can occur when implicitly casting numerical data types of a larger size, into a variable with a data type of a smaller size. This forces the program to discard some bits of information from the number. Depending on how the numerical data types are stored in memory, this is often the bits with the highest value, causing substantial corruption of the stored number. Alternatively, the sign bit of a signed integer could be lost, completely reversing the intention of the number.

General Recommendations

How to avoid it

- Avoid casting larger data types to smaller types.
- Prefer promoting the target variable to a large enough data type.
- If downcasting is necessary, always check that values are valid and in range of the target type, before casting

---

Source Code Examples

Dangerous Functions

Risk

What might happen

Use of dangerous functions may expose varying risks associated with each particular function, with potential impact of improper usage of these functions varying significantly. The presence of such functions indicates a flaw in code maintenance policies and adherence to secure coding practices, in a way that has allowed introducing known dangerous code into the application.

Cause

How does it happen

A dangerous function has been identified within the code. Functions are often deemed dangerous to use for numerous reasons, as there are different sets of vulnerabilities associated with usage of such functions. For example, some string copy and concatenation functions are vulnerable to Buffer Overflow, Memory Disclosure, Denial of Service and more. Use of these functions is not recommended.

General Recommendations

How to avoid it

- Deploy a secure and recommended alternative to any functions that were identified as dangerous.
    - If no secure alternative is found, conduct further researching and testing to identify whether current usage successfully sanitizes and verifies values, and thus successfully avoids the use-cases for whom the function is indeed dangerous
- Conduct a periodical review of methods that are in use, to ensure that all external libraries and built-in functions are up-to-date and whose use has not been excluded from best secure coding practices.

Source Code Examples

CPP

Buffer Overflow in gets()

```
int main()
{
	char buf[10];

	printf("Please enter your name: ");
	gets(buf); // veryveryverylongname
	if (buf == ACCEPTED_NAME)
	{
		// Do something
	}
	return 0;
}
```

Safe reading from user

```
int main()
{
	char buf[10];

	printf("Please enter your name: ");
	fgets(buf, sizeof(buf), stdin); //setting the amount of bytes to read
	if (buf == ACCEPTED_NAME)
	{
		//Do something
	}
	return 0;
}
```

Unsafe function for string copy

```
int main(int argc, char* argv[])
{
	char buf[10];
	strcpy(buf, argv[1]); // overflow occurs when len(argv[1]) > 10 bytes

	return 0;
}
```

Safe string copy

```
int main(int argc, char* argv[])
{
	char buf[10];
	strncpy(buf, argv[1], sizeof(buf));
	buf[9]= '\0'; //strncpy doesn't NULL terminates

	return 0;
}
```

Unsafe format string

```
int main(int argc, char* argv[])
{
	printf(argv[1]); // If argv[1] contains a format token, such as %s,%x or %d, will cause
an access violation
	return 0;
}
```

Safe format string

```
int main(int argc, char* argv[])
{
      printf("%s", argv[1]); // Second parameter is not a formattable string
      return 0;
}
```

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 After Free

Risk

What might happen

A use after free error will cause code to use an area of memory previously assigned with a specific value, which has since been freed and may have been overwritten by another value. This error will likely cause unexpected behavior, memory corruption and crash errors. In some cases where the freed and used section of memory is used to determine execution flow, and the error can be induced by an attacker, this may result in execution of malicious code.

Cause

How does it happen

Pointers to variables allow code to have an address with a set size to a dynamically allocated variable. Eventually, the pointer's destination may become free - either explicitly in code, such as when programmatically freeing this variable, or implicitly, such as when a local variable is returned - once it is returned, the variable's scope is released. Once freed, this memory will be re-used by the application, overwritten with new data. At this point, dereferencing this pointer will potentially resolve newly written and unexpected data.

General Recommendations

How to avoid it

- Do not return local variables or pointers
- Review code to ensure no flow allows use of a pointer after it has been explicitly freed

---

Source Code Examples

CPP

Use of Variable after It was Freed

```
free(input);
printf("%s", input);
```

Use of Pointer to Local Variable That Was Freed On Return

```
int* func1()
{
	int i;
	i = 1;
	return &i;
}

void func2()

{

	int j;
	j = 5;
```

```
}

//..
	int * i = func1();
	printf("%d\r\n", *i); // Output could be 1 or Segmentation Fault
	func2();
	printf("%d\r\n", *i); // Output is 5, which is j's value, as func2() overwrote data in
the stack
//..
```

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());
```

Uncontrolled Recursion

Weakness ID: 674 *(Weakness Base)* **Status:** Draft

Description

Description Summary

The product does not properly control the amount of recursion that takes place, which consumes excessive resources, such as allocated memory or the program stack.

Alternate Terms

Stack Exhaustion

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Languages

All

Common Consequences

Scope	Effect
Availability	Resources including CPU, memory, and stack memory could be rapidly consumed or exhausted, eventually leading to an exit or crash.
Confidentiality	In some cases, an application's interpreter might kill a process or thread that appears to be consuming too much resources, such as with PHP's memory_limit setting. When the interpreter kills the process/thread, it might report an error containing detailed information such as the application's installation path.

Observed Examples

Reference	Description
CVE-2007-1285	Deeply nested arrays trigger stack exhaustion.
CVE-2007-3409	Self-referencing pointers create infinite loop and resultant stack exhaustion.

Potential Mitigations

Limit the number of recursive calls to a reasonable number.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	361	Time and State	**Development Concepts (primary)699**
ChildOf	Weakness Class	691	Insufficient Control Flow Management	**Research Concepts (primary)1000**
ChildOf	Category	730	OWASP Top Ten 2004 Category A9 - Denial of Service	**Weaknesses in OWASP Top Ten (2004) (primary)711**

Affected Resources

- CPU

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
OWASP Top Ten 2004	A9	CWE More Specific	Denial of Service

Related Attack Patterns

CAPEC-ID	Attack Pattern Name	(CAPEC Version: 1.5)
82	Violating Implicit Assumptions Regarding XML Content (aka XML Denial of Service (XDoS))	
99	XML Parser Attack	

Content History

Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Potential Mitigations, Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Common Consequences, Relationships, Taxonomy Mappings		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Related Attack Patterns		

BACK TO TOP

Use of Function with Inconsistent Implementations

Weakness ID: 474 *(Weakness Base)* **Status:** Draft

Description

Description Summary

The code uses a function that has inconsistent implementations across operating systems and versions, which might cause security-relevant portability problems.

Time of Introduction

- Architecture and Design
- Implementation

Applicable Platforms

Languages

C: *(Often)*

PHP: *(Often)*

All

Potential Mitigations

Do not accept inconsistent behavior from the API specifications when the deviant behavior increase the risk level.

Other Notes

The behavior of functions in this category varies by operating system, and at times, even by operating system version. Implementation differences can include:

- Slight differences in the way parameters are interpreted leading to inconsistent results.
- Some implementations of the function carry significant security risks.
- The function might not be defined on all platforms.

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Weakness Class	398	Indicator of Poor Code Quality	**Development Concepts (primary)699 Seven Pernicious Kingdoms (primary)700 Research Concepts (primary)1000**
ParentOf	Weakness Variant	589	Call to Non-ubiquitous API	**Research Concepts (primary)1000**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
7 Pernicious Kingdoms			Inconsistent Implementations

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	7 Pernicious Kingdoms		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Potential Mitigations, Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Relationships, Other Notes, Taxonomy Mappings		
Previous Entry Names			
Change Date	**Previous Entry Name**		
2008-04-11	Inconsistent Implementations		

BACK TO TOP

Potential Off by One Error in Loops

Risk

What might happen

An off by one error may result in overwriting or over-reading of unintended memory; in most cases, this can result in unexpected behavior and even application crashes. In other cases, where allocation can be controlled by an attacker, a combination of variable assignment and an off by one error can result in execution of malicious code.

Cause

How does it happen

Often when designating variables to memory, a calculation error may occur when determining size or length that is off by one.

For example in loops, when allocating an array of size 2, its cells are counted as 0,1 - therefore, if a For loop iterator on the array is incorrectly set with the start condition i=0 and the continuation condition i<=2, three cells will be accessed instead of 2, and an attempt will be made to write or read cell [2], which was not originally allocated, resulting in potential corruption of memory outside the bounds of the originally assigned array.

Another example occurs when a null-byte terminated string, in the form of a character array, is copied without its terminating null-byte. Without the null-byte, the string representation is unterminated, resulting in certain functions to over-read memory as they expect the missing null terminator.

General Recommendations

How to avoid it

- Always ensure that a given iteration boundary is correct:
  - With array iterations, consider that arrays begin with cell 0 and end with cell n-1, for a size n array.
  - With character arrays and null-byte terminated string representations, consider that the null byte is required and should not be overwritten or ignored; ensure functions in use are not vulnerable to off-by-one, specifically for instances where null-bytes are automatically appended after the buffer, instead of in place of its last character.
- Where possible, use safe functions that manage memory and are not prone to off-by-one errors.

Source Code Examples

CPP

Off-By-One in For Loop

```
int *ptr;
ptr = (int*)malloc(5 * sizeof(int));
for (int i = 0; i <= 5; i++)
{
      ptr[i] = i * 2 + 1; // ptr[5] will be set, but is out of bounds
```

```
}
```

Proper Iteration in For Loop

```
int *ptr;
ptr = (int*)malloc(5 * sizeof(int));
for (int i = 0; i < 5; i++)
{
      ptr[i] = i * 2 + 1; // ptr[0-4] are well defined
}
```

Off-By-One in strncat

```
strncat(buf, input, sizeof(buf) - strlen(buf)); // actual value should be sizeof(buf)-
strlen(buf)-1 - this form will overwrite the terminating nullbyte
```

Potential Precision Problem

Risk

What might happen

Buffer overflow attacks, in their various forms, could allow an attacker to control certain areas of memory. Typically, this is used to overwrite data on the stack necessary for the program to function properly, such as code and memory addresses, though other forms of this attack exist. Exploiting this vulnerability can generally lead to system crashes, infinite loops, or even execution of arbitrary code.

Cause

How does it happen

Buffer Overflows can manifest in numerous different variations. In 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

Indicator of Poor Code Quality

Weakness ID: 398 *(Weakness Class)* **Status:** Draft

Description

Description Summary

The code has features that do not directly introduce a weakness or vulnerability, but indicate that the product has not been carefully developed or maintained.

Extended Description

Programs are more likely to be secure when good development practices are followed. If a program is complex, difficult to maintain, not portable, or shows evidence of neglect, then there is a higher likelihood that weaknesses are buried in the code.

Time of Introduction

- Architecture and Design
- Implementation

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	18	Source Code	**Development Concepts (primary)699**
ChildOf	Weakness Class	710	Coding Standards Violation	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	107	Struts: Unused Validation Form	**Research Concepts (primary)1000**
ParentOf	Weakness Variant	110	Struts: Validator Without Form Field	**Research Concepts (primary)1000**
ParentOf	Category	399	Resource Management Errors	**Development Concepts (primary)699**
ParentOf	Weakness Base	401	Failure to Release Memory Before Removing Last Reference ('Memory Leak')	**Seven Pernicious Kingdoms (primary)700**
ParentOf	Weakness Base	404	Improper Resource Shutdown or Release	Development Concepts699 **Seven Pernicious Kingdoms (primary)700**
ParentOf	Weakness Variant	415	Double Free	**Seven Pernicious Kingdoms (primary)700**
ParentOf	Weakness Base	416	Use After Free	**Seven Pernicious Kingdoms (primary)700**
ParentOf	Weakness Variant	457	Use of Uninitialized Variable	**Seven Pernicious Kingdoms (primary)700**
ParentOf	Weakness Base	474	Use of Function with Inconsistent Implementations	**Development Concepts (primary)699 Seven Pernicious Kingdoms (primary)700 Research Concepts (primary)1000**
ParentOf	Weakness Base	475	Undefined Behavior for Input to API	**Development Concepts (primary)699 Seven Pernicious Kingdoms (primary)700**
ParentOf	Weakness Base	476	NULL Pointer	**Development**

			Dereference	**Concepts (primary)699 Seven Pernicious Kingdoms (primary)700 Research Concepts (primary)1000**
ParentOf	Weakness Base	477	Use of Obsolete Functions	**Development Concepts (primary)699 Seven Pernicious Kingdoms (primary)700 Research Concepts (primary)1000**
ParentOf	Weakness Variant	478	Missing Default Case in Switch Statement	**Development Concepts (primary)699**
ParentOf	Weakness Variant	479	Unsafe Function Call from a Signal Handler	**Development Concepts (primary)699**
ParentOf	Weakness Variant	483	Incorrect Block Delimitation	**Development Concepts (primary)699**
ParentOf	Weakness Base	484	Omitted Break Statement in Switch	**Development Concepts (primary)699** Research Concepts1000
ParentOf	Weakness Variant	546	Suspicious Comment	**Development Concepts (primary)699 Research Concepts (primary)1000**
ParentOf	Weakness Variant	547	Use of Hard-coded, Security-relevant Constants	**Development Concepts (primary)699 Research Concepts (primary)1000**
ParentOf	Weakness Variant	561	Dead Code	**Development Concepts (primary)699 Research Concepts (primary)1000**
ParentOf	Weakness Base	562	Return of Stack Variable Address	**Development Concepts (primary)699** Research Concepts1000
ParentOf	Weakness Variant	563	Unused Variable	**Development Concepts (primary)699 Research Concepts (primary)1000**
ParentOf	Category	569	Expression Issues	**Development Concepts (primary)699**
ParentOf	Weakness Variant	585	Empty Synchronized Block	**Development Concepts (primary)699 Research Concepts (primary)1000**
ParentOf	Weakness Variant	586	Explicit Call to Finalize()	**Development Concepts (primary)699**
ParentOf	Weakness Variant	617	Reachable Assertion	**Development Concepts (primary)699**
ParentOf	Weakness Base	676	Use of Potentially Dangerous Function	**Development Concepts (primary)699 Research Concepts (primary)1000**
MemberOf	View	700	Seven Pernicious Kingdoms	**Seven Pernicious Kingdoms (primary)700**

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name

7 Pernicious Kingdoms			Code Quality

Content History

Submissions			
Submission Date	**Submitter**	**Organization**	**Source**
	7 Pernicious Kingdoms		Externally Mined
Modifications			
Modification Date	**Modifier**	**Organization**	**Source**
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Description, Relationships, Taxonomy Mappings		
2009-10-29	CWE Content Team	MITRE	Internal
	updated Relationships		
Previous Entry Names			
Change Date	**Previous Entry Name**		
2008-04-11	Code Quality		

BACK TO TOP

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

Use of Insufficiently Random Values

Risk

What might happen

Random values are often used as a mechanism to prevent malicious users from guessing a value, such as a password, encryption key, or session identifier. Depending on what this random value is used for, an attacker would be able to predict the next numbers generated, or previously generated values. This could enable the attacker to hijack another user's session, impersonate another user, or crack an encryption key (depending on what the pseudo-random value was used for).

Cause

How does it happen

The application uses a weak method of generating pseudo-random values, such that other numbers could be determined from a relatively small sample size. Since the pseudo-random number generator used is designed for statistically uniform distribution of values, it is approximately deterministic. Thus, after collecting a few generated values (e.g. by creating a few individual sessions, and collecting the sessionids), it would be possible for an attacker to calculate another sessionid.
Specifically, if this pseudo-random value is used in any security context, such as passwords, keys, or secret identifiers, an attacker would be able to predict the next numbers generated, or previously generated values.

General Recommendations

How to avoid it

Generic Guidance:

- Whenever unpredicatable numbers are required in a security context, use a cryptographically strong random number generator, instead of a statistical pseudo-random generator.
- Use the cryptorandom generator that is built-in to your language or platform, and ensure it is securely seeded. Do not seed the generator with a weak, non-random seed. (In most cases, the default is securely random).
- Ensure you use a long enough random value, to make brute-force attacks unfeasible.

Specific Recommendations:

- Do not use the statistical pseudo-random number generator, use the cryptorandom generator instead. In Java, this is the SecureRandom class.

Source Code Examples

Java

Use of a weak pseudo-random number generator

```
Random random = new Random();

long sessNum = random.nextLong();

String sessionId = sessNum.toString();
```

Cryptographically secure random number generator

```
SecureRandom random = new SecureRandom();

byte sessBytes[] = new byte[32];

random.nextBytes(sessBytes);

String sessionId = new String(sessBytes);
```

Objc

Use of a weak pseudo-random number generator

```
long sessNum = rand();
NSString* sessionId = [NSString stringWithFormat:@"%ld", sessNum];
```

Cryptographically secure random number generator

```
UInt32 sessBytes;
SecRandomCopyBytes(kSecRandomDefault, sizeof(sessBytes), (uint8_t*)&sessBytes);

NSString* sessionId = [NSString stringWithFormat:@"%llu", sessBytes];
```

Swift

Use of a weak pseudo-random number generator

```
let sessNum = rand();
let sessionId = String(format:"%ld", sessNum)
```

Cryptographically secure random number generator

```
var sessBytes: UInt32 = 0
withUnsafeMutablePointer(&sessBytes, { (sessBytesPointer) -> Void in
    let castedPointer = unsafeBitCast(sessBytesPointer, UnsafeMutablePointer<UInt8>.self)
    SecRandomCopyBytes(kSecRandomDefault, sizeof(UInt32), castedPointer)
})

let sessionId = String(format:"%llu", sessBytes)
```

Unchecked Return Value

Risk

What might happen

A program that does not check function return values could cause the application to enter an undefined state. This could lead to unexpected behavior and unintended consequences, including inconsistent data, system crashes or other error-based exploits.

Cause

How does it happen

The application calls a system function, but does not receive or check the result of this funciton. These functions often return error codes in the result, or share other status codes with it's caller. The application simply ignores this result value, losing this vital information.

General Recommendations

How to avoid it

- Always check the result of any called function that returns a value, and verify the result is an expected value.

- Ensure the calling function responds to all possible return values.

- Expect runtime errors and handle them gracefully. Explicitly define a mechanism for handling unexpected errors.

Source Code Examples

CPP

Unchecked Memory Allocation

```
buff = (char*) malloc(size);
strncpy(buff, source, size);
```

Safer Memory Allocation

```
buff = (char*) malloc(size+1);
if (buff==NULL) exit(1);

strncpy(buff, source, size);
buff[size] = '\0';
```

Use of sizeof() on a Pointer Type

Weakness ID: 467 *(Weakness Variant)* **Status:** Draft

Description

Description Summary

The code calls sizeof() on a malloced pointer type, which always returns the wordsize/8. This can produce an unexpected result if the programmer intended to determine how much memory has been allocated.

Time of Introduction

- Implementation

Applicable Platforms

Languages

C

C++

Common Consequences

Scope	Effect
Integrity	This error can often cause one to allocate a buffer that is much smaller than what is needed, leading to resultant weaknesses such as buffer overflows.

Likelihood of Exploit

High

Demonstrative Examples

Example 1

Care should be taken to ensure sizeof returns the size of the data structure itself, and not the size of the pointer to the data structure.

In this example, sizeof(foo) returns the size of the pointer.

(Bad Code)

Example Languages: **C and C++**

```
double *foo;
...
foo = (double *)malloc(sizeof(foo));
```

In this example, sizeof(*foo) returns the size of the data structure and not the size of the pointer.

(Good Code)

Example Languages: **C and C++**

```
double *foo;
...
foo = (double *)malloc(sizeof(*foo));
```

Example 2

This example defines a fixed username and password. The AuthenticateUser() function is intended to accept a username and a password from an untrusted user, and check to ensure that it matches the username and password. If the username and password match, AuthenticateUser() is intended to indicate that authentication succeeded.

(Bad Code)

```
/* Ignore CWE-259 (hard-coded password) and CWE-309 (use of password system for authentication) for this example. */

char *username = "admin";
char *pass = "password";

int AuthenticateUser(char *inUser, char *inPass) {
```

```
printf("Sizeof username = %d\n", sizeof(username));
printf("Sizeof pass = %d\n", sizeof(pass));

if (strncmp(username, inUser, sizeof(username))) {
printf("Auth failure of username using sizeof\n");
return(AUTH_FAIL);
}
/* Because of CWE-467, the sizeof returns 4 on many platforms and architectures. */
if (! strncmp(pass, inPass, sizeof(pass))) {
printf("Auth success of password using sizeof\n");
return(AUTH_SUCCESS);
}
else {
printf("Auth fail of password using sizeof\n");
return(AUTH_FAIL);
}
}

int main (int argc, char **argv)
{
int authResult;

if (argc < 3) {
ExitError("Usage: Provide a username and password");
}
authResult = AuthenticateUser(argv[1], argv[2]);
if (authResult != AUTH_SUCCESS) {
ExitError("Authentication failed");
}
else {
DoAuthenticatedTask(argv[1]);
}
}
```

In AuthenticateUser(), because sizeof() is applied to a parameter with an array type, the sizeof() call might return 4 on many modern architectures. As a result, the strncmp() call only checks the first four characters of the input password, resulting in a partial comparison (CWE-187), leading to improper authentication (CWE-287).

Because of the partial comparison, any of these passwords would still cause authentication to succeed for the "admin" user:

(Attack)

```
pass5
passABCDEFGH
passWORD
```

Because only 4 characters are checked, this significantly reduces the search space for an attacker, making brute force attacks more feasible.

The same problem also applies to the username, so values such as "adminXYZ" and "administrator" will succeed for the username.

Potential Mitigations

Phase: Implementation

Use expressions such as "sizeof(*pointer)" instead of "sizeof(pointer)", unless you intend to run sizeof() on a pointer type to gain some platform independence or if you are allocating a variable on the stack.

Other Notes

The use of sizeof() on a pointer can sometimes generate useful information. An obvious case is to find out the wordsize on a platform. More often than not, the appearance of sizeof(pointer) indicates a bug.

Weakness Ordinalities

Ordinality	Description
Primary	*(where the weakness exists independent of other weaknesses)*

Relationships

Nature	Type	ID	Name	View(s) this relationship pertains to
ChildOf	Category	465	Pointer Issues	**Development Concepts (primary)699**
ChildOf	Weakness Class	682	Incorrect Calculation	**Research Concepts (primary)1000**
ChildOf	Category	737	CERT C Secure Coding Section 03 - Expressions (EXP)	**Weaknesses Addressed by the CERT C Secure Coding Standard (primary)734**
ChildOf	Category	740	CERT C Secure Coding Section 06 - Arrays (ARR)	Weaknesses Addressed by the CERT C Secure Coding Standard734
CanPrecede	Weakness Base	131	Incorrect Calculation of Buffer Size	Research Concepts1000

Taxonomy Mappings

Mapped Taxonomy Name	Node ID	Fit	Mapped Node Name
CLASP			Use of sizeof() on a pointer type
CERT C Secure Coding	ARR01-C		Do not apply the sizeof operator to a pointer when taking the size of an array
CERT C Secure Coding	EXP01-C		Do not take the size of a pointer to determine the size of the pointed-to type

White Box Definitions

A weakness where code path has:

1. end statement that passes an identity of a dynamically allocated memory resource to a sizeof operator
2. start statement that allocates the dynamically allocated memory resource

References

Robert Seacord. "EXP01-A. Do not take the sizeof a pointer to determine the size of a type". <https://www.securecoding.cert.org/confluence/display/seccode/EXP01-A.+Do+not+take+the+sizeof+a+pointer+to+determine+the+size+of+a+type>.

Content History

Submissions

Submission Date	Submitter	Organization	Source
	CLASP		Externally Mined

Modifications

Modification Date	Modifier	Organization	Source
2008-07-01	Eric Dalci	Cigital	External
	updated Time of Introduction		
2008-08-01		KDM Analytics	External
	added/updated white box definitions		
2008-09-08	CWE Content Team	MITRE	Internal
	updated Applicable Platforms, Common Consequences, Relationships, Other Notes, Taxonomy Mappings, Weakness Ordinalities		
2008-11-24	CWE Content Team	MITRE	Internal
	updated Relationships, Taxonomy Mappings		
2009-03-10	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2009-12-28	CWE Content Team	MITRE	Internal
	updated Demonstrative Examples		
2010-02-16	CWE Content Team	MITRE	Internal
	updated Relationships		

BACK TO TOP

NULL Pointer Dereference

Risk

What might happen

A null pointer dereference is likely to cause a run-time exception, a crash, or other unexpected behavior.

Cause

How does it happen

Variables which are declared without being assigned will implicitly retain a null value until they are assigned. The null value can also be explicitly set to a variable, to ensure clear out its contents. Since null is not really a value, it may not have object variables and methods, and any attempt to access contents of a null object, instead of verifying it is set beforehand, will result in a null pointer dereference exception.

General Recommendations

How to avoid it

- For any variable that is created, ensure all logic flows between declaration and use assign a non-null value to the variable first.
- Enforce null checks on any received variable or object before it is dereferenced, to ensure it does not contain a null assigned to it elsewhere.
- Consider the need to assign null values in order to overwrite initialized variables. Consider reassigning or releasing these variables instead.

Source Code Examples

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

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--;
       }
    }
}
```

Scanned Languages

Language	Hash Number	Change Date
CPP	4541647240435660	6/19/2024
Common	0105849645654507	6/19/2024