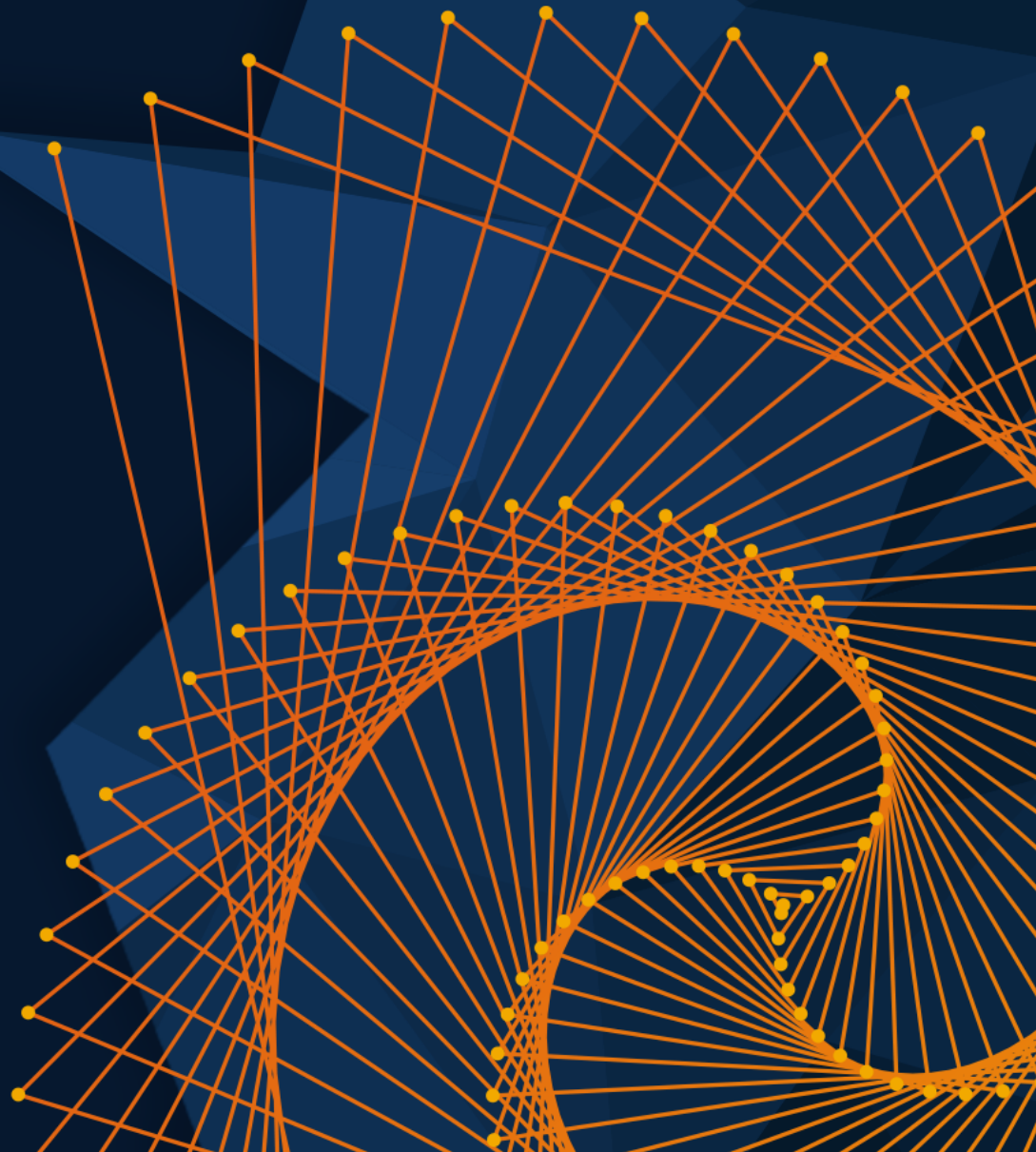
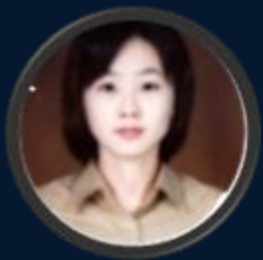


MATLAB EXPO

2024.06.11 | 그랜드 인터컨티넨탈 서울 파르나스

Check Cybersecurity Coding Rules using Polyspace Bug Finder

Shinae Lee, HL Mando



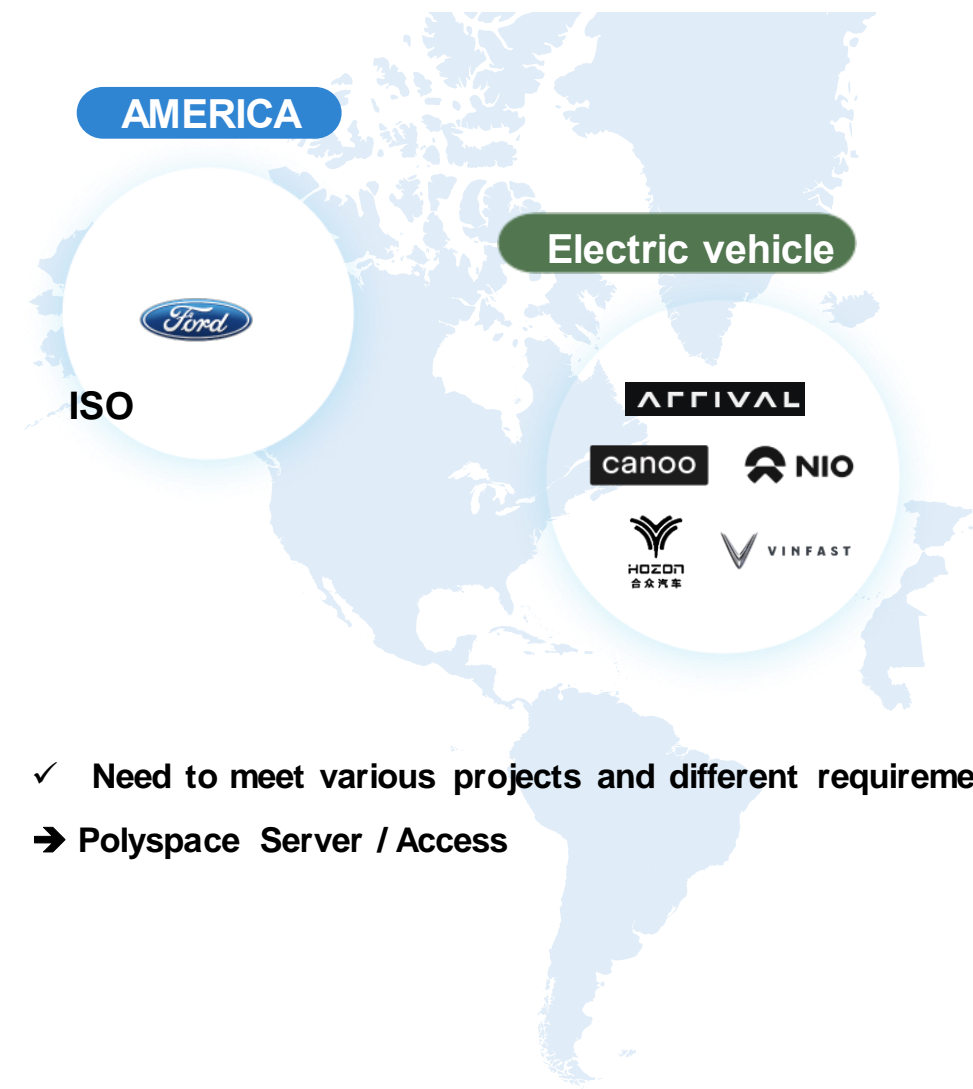
Agenda

- Why does HL Mando need Cyber Secure Coding Guidelines?
- HL Mando Cyber Security Coding Rules Using Polyspace Bug Finder
- Future Plan & Conclusion

Why does HL Mando need Cyber Secure Coding Guidelines?

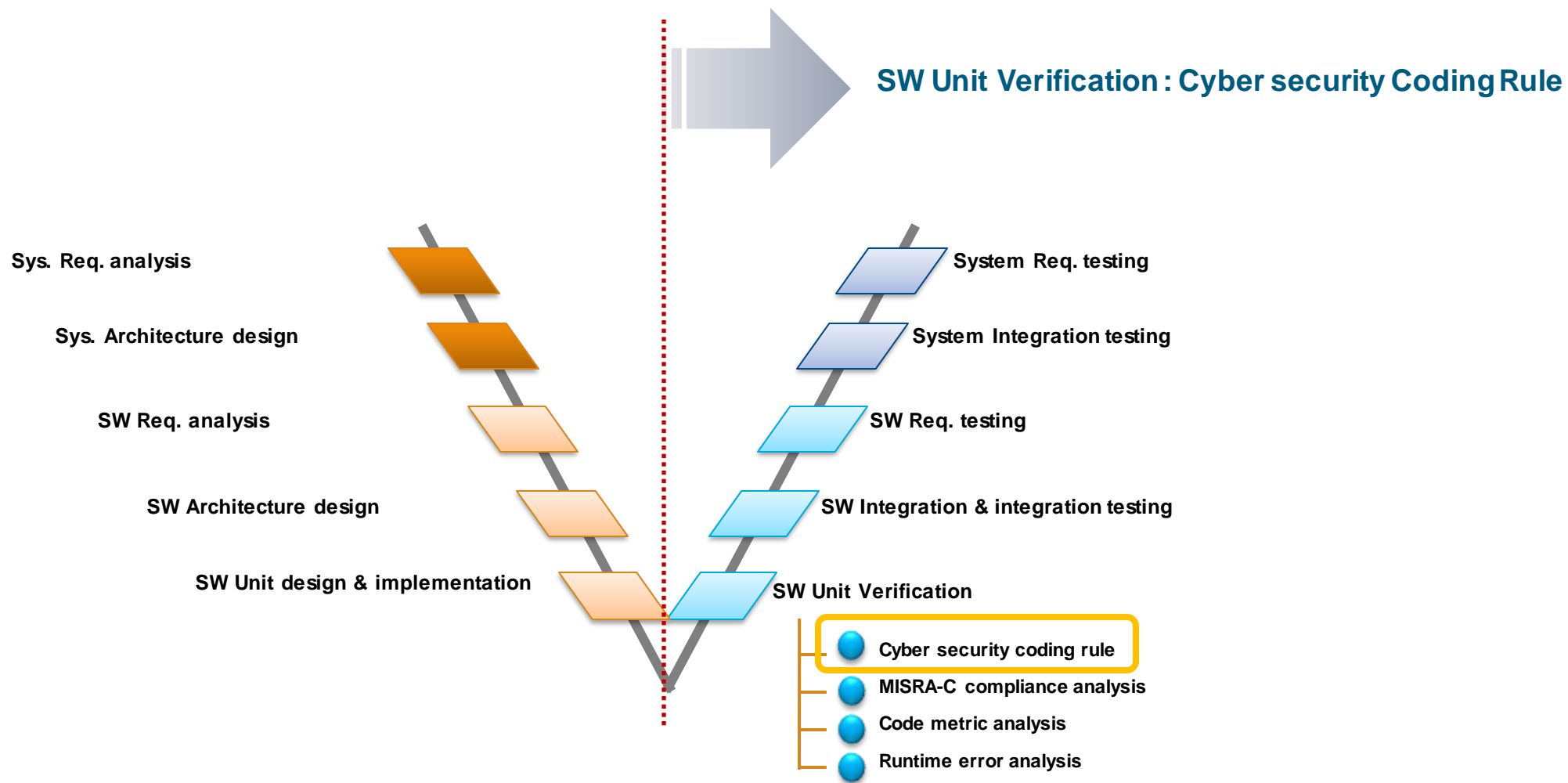
- ✓ Strengthen safety requirements and regulations : UN Regulation No.155
- ✓ Cyber Security Management System (CSMS) certificate is required
- ✓ One of the main items of CSMS includes verification of the cybersecurity management system of parts companies.
- ✓ One of the verification items is cyber secure coding.
- ✓ Increasing market and customer software requirements

Cyber security coding project

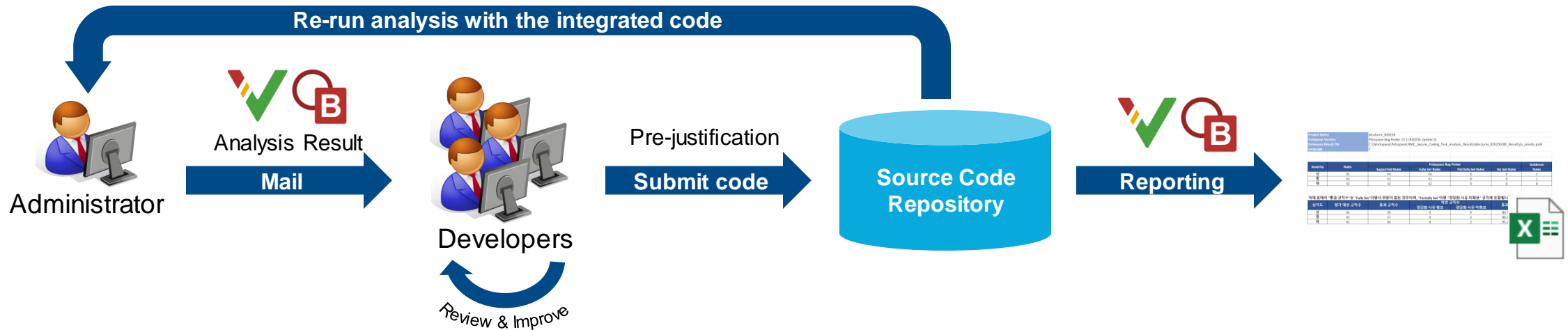


- ✓ Need to meet various projects and different requirements
- ➔ Polyspace Server / Access

Static Verification



Challenges to comply with Cyber Secure Coding



- Administrator **manually** performs the analysis and distributes the result to developers by e-mail
- Each developer **manually** runs the analysis only for their own code on the developer's computer, and each developer adds **pre-justification** in a code
- Administrator manually performs the analysis and distribute it **repeatedly**

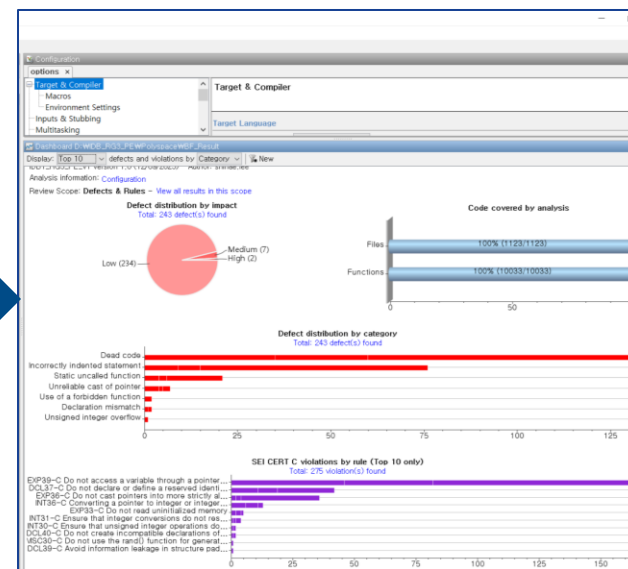
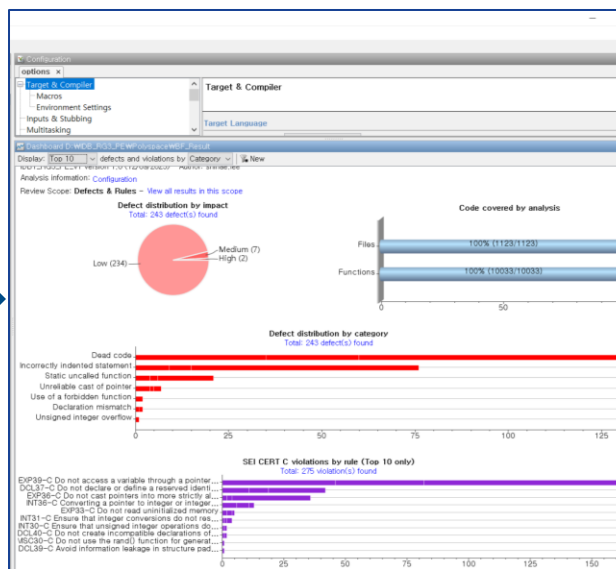
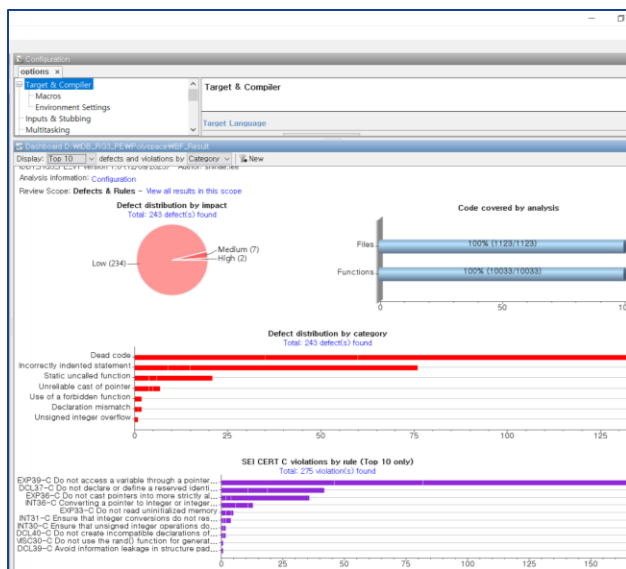
Verification using Polyspace as You Code



Analysis by administrator

Review & Improve by developers

Re-analysis by administrator & developers



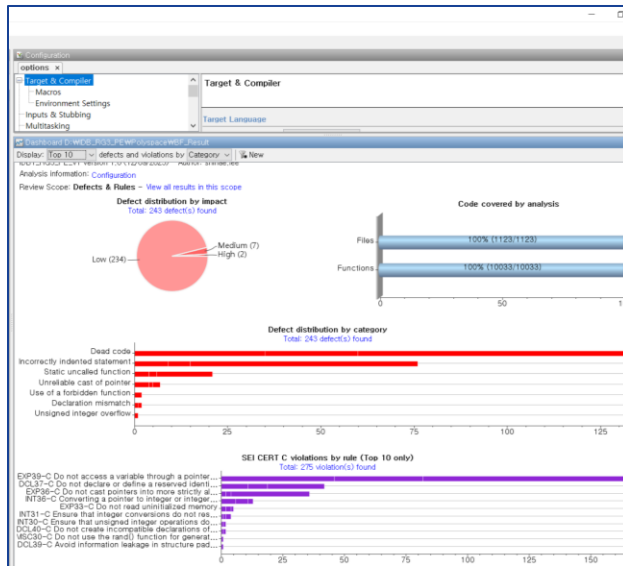
Verification using Polyspace as You Code



Analysis by administrator

Review & Improve by developers

Re-analysis by administrator & developers



Polyspace - Analysis Options: Analysis of Files On Save. Start analysis when saving a file in the Quality Monitoring list.

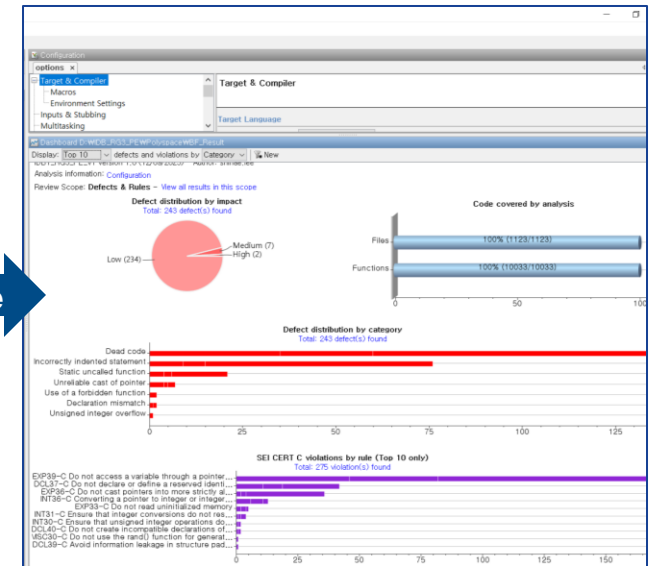
Polyspace - Analysis Options: Analysis Setup. Use a script to configure and run analysis (in setting scriptFile) or specify each setting explicitly (in settings group Manual Setup).

Manual setup: Build. Choose how to extract build options. Use settings buildCommand, buildTask, jsonCompilerDatabaseFile and polyspaceBuildOptionsFile. 'buildCommand', 'buildTask' and 'jsonCompilerDatabaseFile' will need to later run Polyspace: Generate Build Options.

Get from Polyspace build options file.

```

MgdDrv_Input.c
...
return CRC_out;
...
}
...
inline void MgdDrv_setSpiPortPinReg_Direct(Ifx_p_t *
...
static uint32_t init_val;
uint8_t temp_PC;
    
```



Polyspace를 활용한 현대자동차 보안 코딩 가이드 준수 방안
<https://content.mathworks.com/viewer/63365e0b28b33be6d988e7d3>

Cyber Secure Coding analysis on CI(Continuous Integration)



Jenkins



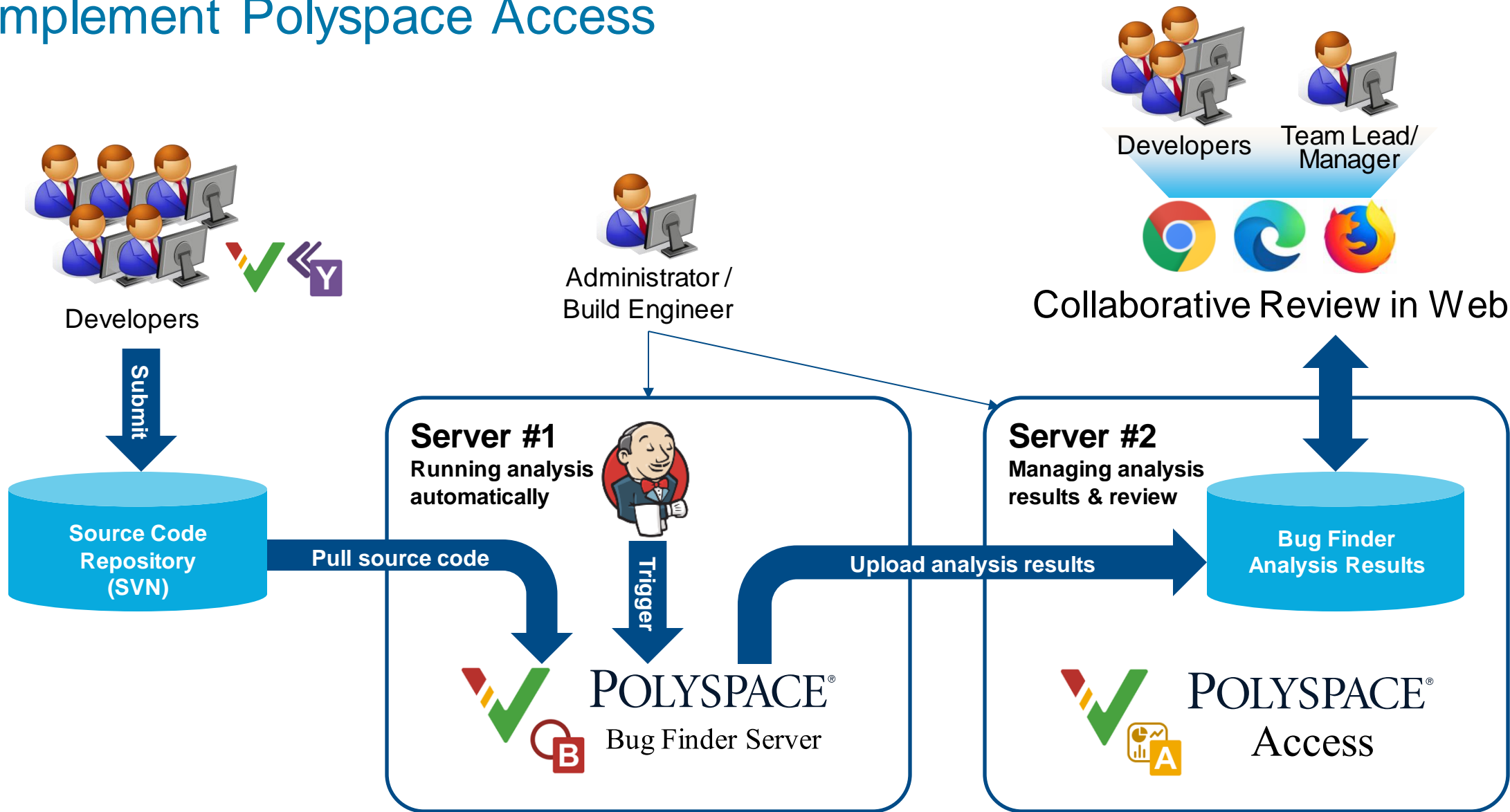
POLYSPACE[®]
Bug Finder Server



POLYSPACE[®]
Access

- CI : Jenkins
- Analysis : Polyspace Bug Finder Server
- Review in Web: Polyspace Access

Implement Polyspace Access



Automated Cyber security coding rule analysis(1)

✓ Polyspace Bug Finder Server

- Set Source Code path
- Analysis option-file opfile [making txt files]

```
-allow-build-error -allow-overwrite -output-options-file options_from_build.txt ../../BuildScript\dbs_build.bat
```

- Create 'options.txt' file with your options

```
Verifying sources compliance ...  
  
Options used with Verifier:  
-prog=IDB2_CEER  
-lang=C  
-polyspace-version=3.8 (R2023a Update 5)  
-verif-version=1.0  
-results-dir=D:\WDBSWrepository\IDB2_Polyspace_CEER\IDB2_CEER_Result  
-date=14/05/2024
```

Automated Cyber security coding rule analysis(2)

- ✓ Polyspace analysis automation for each OEM and HL Mando guidelines

```
Execute Windows batch command ?  
Command  
See the list of available environment variables  
..#.#.#BuildScript#Polyspace_MX5_TMED.bat
```

```
echo Running Polyspace analysis...  
"%bug_finder_command%" -options-file options_from_build.txt -options-file ..\..\PolyspaceScripts\HKMC_Secure_Coding_C_R2023a\Analysis_Automation_Options_For_HMC_Secure_Coding_Guidelines_R2023a.txt -results-dir  
  
echo Running Polyspace analysis...  
"%bug_finder_command%" -options-file options_from_build.txt -options-file ..\..\PolyspaceScripts\HL_Mando_Coding_Rule_Secure_Coding_C_R2023a\Analysis_Automation_Options_For_HMC_Secure_Coding_Guidelines_R2023a  
  
echo Running Polyspace analysis...  
"%bug_finder_command%" -options-file options_from_build.txt -options-file ..\..\PolyspaceScripts\ISO/IEC_TS_17961_Secure_Coding_C_R2023a\Analysis_Automation_Options_For_HMC_Secure_Coding_Guidelines_R2023a.
```

Review findings in Polyspace Access (1)

Polyspace Analysis Report

Polyspace Bug Finder analysis is completed

Analysis Information	
Number of CWE violations	2980
Number of CERT-C violations	6015
Number of Defects	275

You can open more details in Polyspace Access URL at below

Analysis Access URL : ["http://172.20.112.82-9443/metrics/index.html?a=review&p=28&r=17"](http://172.20.112.82-9443/metrics/index.html?a=review&p=28&r=17)

You can open Jenkins Job details at below link.

Build : ["http://172.20.112.238-8080/job/IDB1_cyber_Security_Polyspace/job/HMC_JX1_PE/job/HMC_JX1_PE_Polyspace_TEST/103/"](http://172.20.112.238-8080/job/IDB1_cyber_Security_Polyspace/job/HMC_JX1_PE/job/HMC_JX1_PE_Polyspace_TEST/103/)

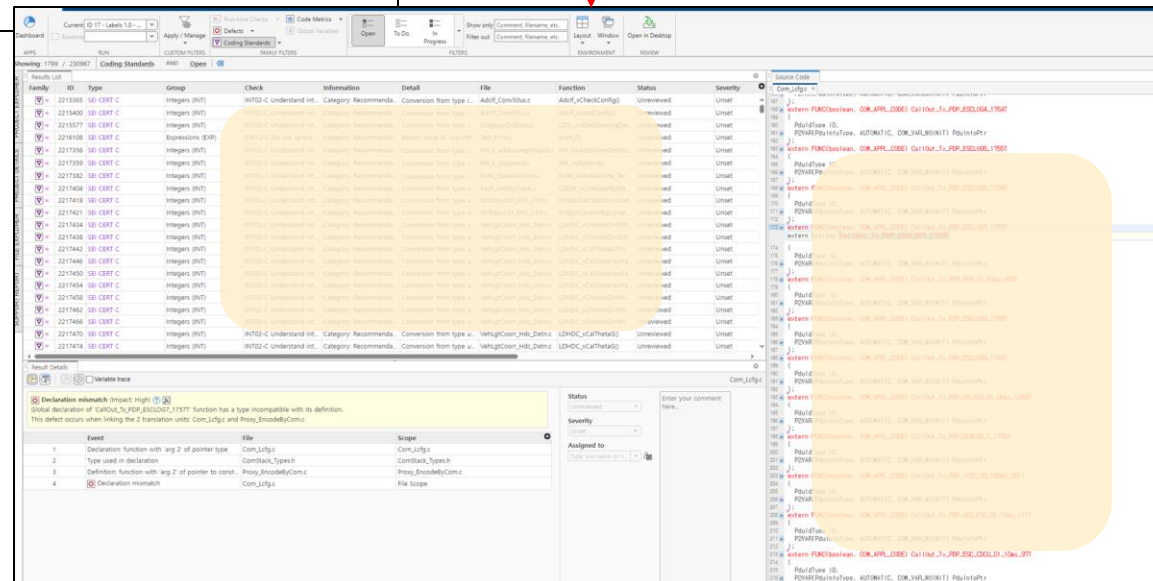
ADMIN Information	
ADMIN	ADMIN
SW1 ACCESS LOGIN	
ID	shinae.lee



POLYSPACE

Sign in to your account

Forgot password?



The screenshot shows the Polyspace Access interface with a list of defects. A red arrow points from the URL in the report to the interface. A yellow highlight is on a specific defect row in the table.

Family ID	Type	Group	Check	Information	Detail	File	Function	Status	Severity
2215385	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Adapt_ConvSubst	Adapt_CheckConfig	Unreviewed	Unset
2215400	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Adapt_ConvSubst	Adapt_CheckConfig	Unreviewed	Unset
2215377	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Adapt_ConvSubst	Adapt_CheckConfig	Unreviewed	Unset
2216108	SE CERT C	Expressions (EXP)	EXP-Do not negate	Category: Recommendations	Return value of operand '...'	Unreviewed	Unset
2217356	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217359	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217382	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217426	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217419	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217421	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217434	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217428	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217442	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217446	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217450	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217454	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217458	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217462	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217466	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217470	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset
2217474	SE CERT C	Integers (SMT)	INT32-Understand int.	Category: Recommendations	Conversion from type '...'	Unreviewed	Unset

Result Details

Declaration mismatch (Impact: high)

Global declaration of `Com_Lfgr` for function `Com_Lfgr` has a type incompatible with its definition. This defect occurs when linking the 2 translation units `Com_Lfgr` and `Proxy_EncodedComC`.

Event	File	Scope
1	Declaration function with arg 2 of pointer type	Com_Lfgr
2	Type used in declaration	Com_Lfgr
3	Definition function with arg 2 of pointer to const.	Proxy_EncodedComC
4	Declaration mismatch	Com_Lfgr

Review findings in Polyspace Access (2)

Detail Result

Family	ID	Type	Group	Check	Information	Detail	File
▼ *	2217450	SEI CERT C	Integers (INT)	INTO2-C Understand int...	Category: Recommenda...	Conversion from type u...	VehLgtCoorr_Hdc_Detn...
▼ *	2217454	SEI CERT C	Integers (INT)	INTO2-C Understand int...	Category: Recommenda...	Conversion from type u...	VehLgtCoorr_Hdc_Detn...
▼ *	2217458	SEI CERT C	Integers (INT)	INTO2-C Understand int...	Category: Recommenda...	Conversion from type u...	VehLgtCoorr_Hdc_Detn...
▼ *	2217462	SEI CERT C	Integers (INT)	INTO2-C Understand int...	Category: Recommenda...	Conversion from type u...	VehLgtCoorr_Hdc_Detn...

Rule Violation

Detail Comment

SEI CERT C INTO2-C (Recommendation)
 Understand integer conversion rules
 Conversion from type *unsigned int 16 bits* to type *int 16 bits* might overflow.
 Additional Info:
 Expected values: [-32768 .. 32767].
 Actual values: [0 .. 32768].
 Risk: Truncation or wrap-around of value to fit destination type might lead to unexpected results.
 Fix: Ensure that the destination type is larger than or same as the source type.

Event	File	Scope
1	if predicate allows the failure with value -3...	VehLgtCoorr_Lib_Fct.c
2	Assignment to local variable 'lclAbsTemp'	VehLgtCoorr_Lib_Fct.c
3	Return of function 'LDVLC_u16Abs'	VehLgtCoorr_Lib_Fct.c
4	Exiting function 'LDVLC_u16Abs'	VehLgtCoorr_Hdc_Detn.c

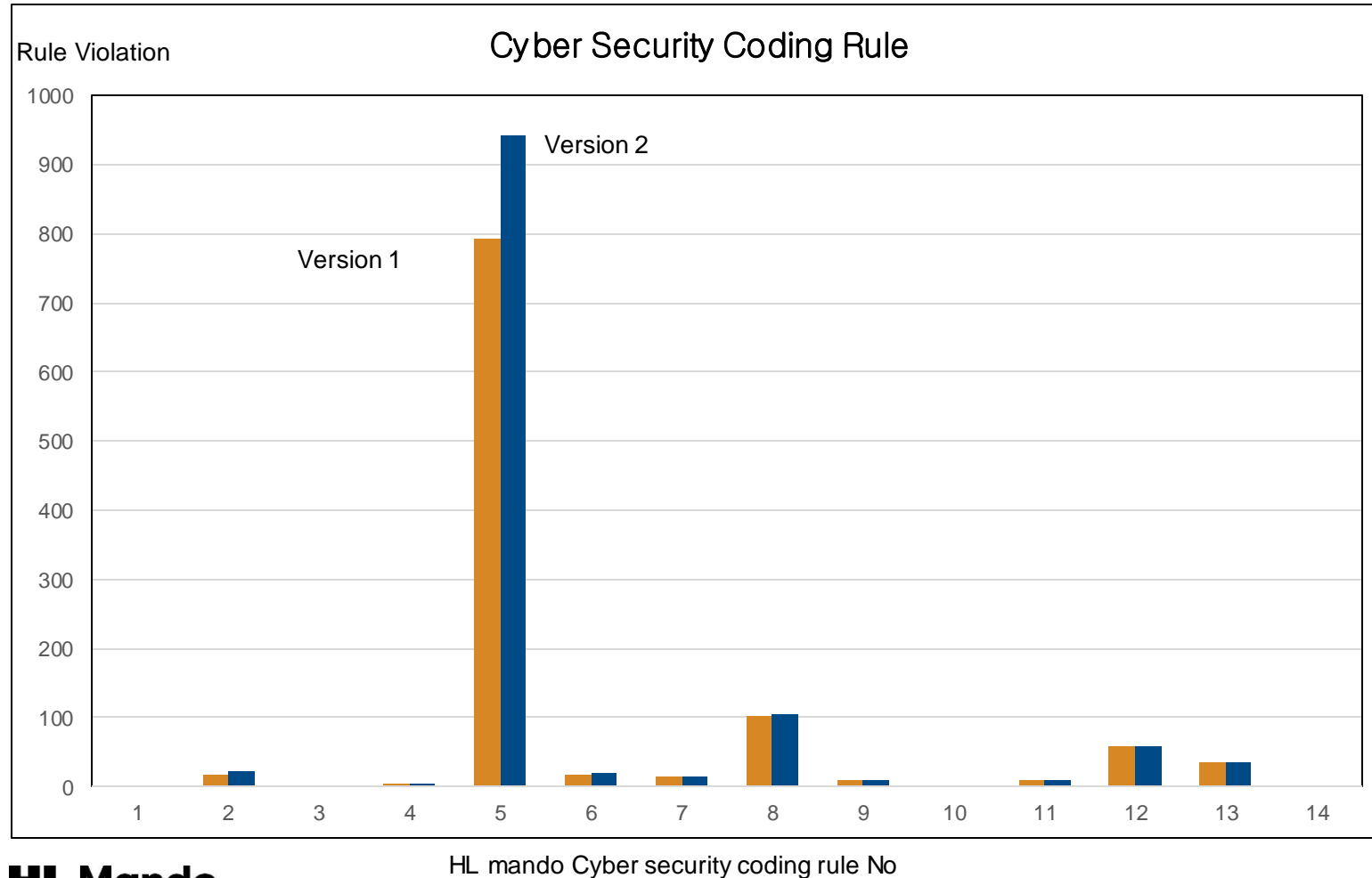
Developer Comment

Violation Code Location

```

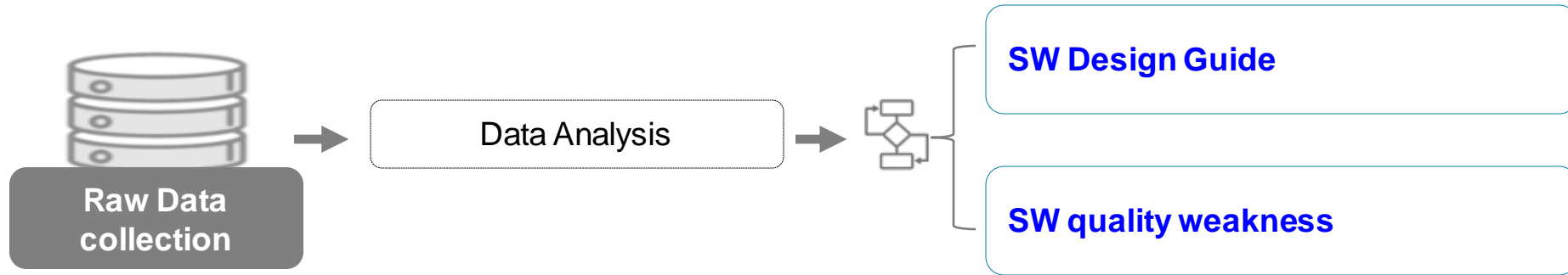
Source Code
Test_Print.c | Com_Lcfc.c | loc_k_mk_addncontrol.c | VehLgtCoorr_Hdc_Detn.c | Com_Core_Read.c
138   lcs16HdcThetaG_diff = lcs16HdcThetaG_F = lcs16HdcThetaG_old;
139   lcs16HdcThetaG_B_diff = lcs16HdcThetaG_B = lcs16HdcThetaG_B_old;
140
141   lcs16_hdc_tempW0 = (sint16)LDVLC_u16Abs(lcs16HdcThetaG_diff)*10;
    
```

Cyber security Coding Rule Violation trend



- ✓ Coding Rule trend analysis whenever SW registered
- ✓ Review and remediation work for next improvement.

Future Plan








< Output >

SW Quality Analysis Results

Cyber security coding : Rule type , violation type

Conclusion

- ✓ Easy to verify cybersecurity coding rules by Polyspace Bug Finder 
- ✓ Access Dashboard globally and check code 
- ✓ Early verification on Developer's PC by Polyspace as You Code 
- ✓ Analyzes the trend of rule violations in cyber secure coding 
- ✓ and improves SW quality 

I want ...

- Run Polyspace Access on Windows
- Automate project creation more smoothly

MATLAB EXPO



© 2024 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

