Issue Report

License Number	Customer
CBD1400346	Nexteer Automotive Corporation Package: GMLAN 3.1 - CANbedded License for GM; MultiChannel Micro: R7F701311 Compiler: GHS 2015.1.7

Maintenance Expiry Date

2014-08-01

SIP Delivery Date	SIP Version
2016-04-29	01.01.35

SLP	Delivery Number
GMLAN 3.1	D01

Report Creation Date

2016-05-06

Contact

In case of questions or the need for an update of the basic software delivery, please contact GMSupport@us.vector.com or your Vector contact person.

Table of Contents

1. Introduction

1.1 Resolving Issues

Reported issues are not necessarily fixed automatically by the next update delivery. If some of the reported issues shall be fixed, please contact Vector to establish an agreement about issues that shall be fixed in upcoming deliveries. Please note that Vector may fix additional issues without explicit request.

1.2 Issue Classification

This Issue Report provides issues that have been detected since the last report. The issues have been classified to facilitate the assessment of their impact:

The chapter 'New Issues' lists issues that have been detected since the last report and which could not be excluded based on the use-case defined in the questionnaire. The issues are classified as follows:

- **Runtime Issues without Workaround:** Runtime issues without a workaround require an update of the basic software delivery in case the issue affects the ECU overall functionality. The effect of an issue to the ECU functionality has to be analyzed by the customer as the basic software usage and its configuration is not known by Vector. The risk of change has also to be taken into account.
- **Runtime Issues with Workaround:** It is not recommended to update a delivery due to a runtime issue with a documented workaround. The effect of an issue to the ECU functionality has to be analyzed by the customer as the basic software usage and its configuration is not known by Vector. The risk of change has also to be taken into account.
- **Apparent Issues:** Apparent issues are detected immediately when using the basic software. If an issue does not show up while working with the basic software the ECU project is not affected by the issue. Apparent issues may or may not have workarounds.
- **Not Released Functionality:** Not released functionalities are modules and features that have not yet passed a complete development cycle (they are e.g. not or only partly tested). For serial production projects the integrator has to ensure that all BETA features are disabled as indicated. If a ESCAN affects a complete BSW module, the module must not be used for serial production.
- **Compiler Warnings:** As a service we report the known compiler warnings. The occurrence of a compiler warning may depend on the used configuration and compiler settings.

The chapter 'New Issues for Information' lists issues that are not relevant for the use case that has been documented in the questionnaire provided to Vector. The issues may, however, be relevant for other use cases. Additionally, issues that have been accepted or are tolerated by the OEM (as defined in the questionnaire) are reported here.

2. New Issues

2.1 Runtime Issues without Workaround

The lists contain issues that have been detected since the last report and which could not be excluded based on the use-cases defined in the questionnaire (see chapter 'New Issues for Information').

2.2 Runtime Issues with Workaround

It is not recommended to update a delivery due to a runtime issue with a documented workaround. The effect of an issue to the ECU functionality has to be analyzed by the customer as the basic software usage and its configuration is not known by Vector. Thereby the risk of change has also to be taken into account.

Index

ESCAN00027894	Memory is overwritten when initializing the CANBedded Stack
Component@Subcomponent:	Nm_Gmlan_Gm@Implementation
First affected version:	3.30.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

Memory is overwritten when initializing the CANBedded Stack.

When does this happen:

The issue occurs always and immediately if CclInitPowerOn or IlInitPowerOn is called with the configuration mentioned below.

In which configuration does this happen:

Any configuration, where the number of Nm Channels differs from the number of Can Channels.

Hint: The generated define in kCanNumberOfChannels in can_cfg.h differs from kNmNumberOfChannels generated to nm_cfg.h

Resolution Description:
Workaround:

Do not call IlInitPowerOn in the application. Instead, call IlInit for each channel which uses the Interaction Layer.

Note for GM ECUs: If the Interaction Layer is not used on the first channel in GENy (channel index 0), the application must additionally call CanInitPowerOn before IlInit.

Example:
The ECU has three CAN channels, where the Interaction Layer is only used on the first two.

IlInit(0); /* Initialize the IL on CAN channel 0 */
IlInit(1); /* Initialize the IL on CAN channel 1 */

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00045854	An incorrect timeout is issued for Flow Control and Consecutive Frame timing supervision.
Component@Subcomponent:	Tp_Iso15765@GenTool_Geny
First affected version:	2.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

An incorrect timeout is issued for Flow Control (TX) and Consecutive Frame (RX) timing supervision in case of large timeouts.

When does this happen:

During runtime at transmission and/or reception of multi frames.

In which configuration does this happen:

This can only appear if channel specific timing is activated (#if defined TP_CHANNEL_SPECIFIC_TIMING)
AND
the configured timeout values are greater than 255 "ticks".
Please note that the number of "ticks" is calculated by dividing the configured timeout value by the configured periodic cycle time of the TP.

Resolution Description:

Workaround:

Use smaller timeouts or increase the call-cycle of the TP task functions.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00068912	Positive response to service $A5 03 not suppressed
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	6.12.01
Fixed in versions:	

Problem Description:

What happens (symptoms):

The positive response to service $A5 03 is not suppressed.
AND possibly
The following compiler warning occurs:

```
 static void DescOemEnableProgrammingMode(DescMsgContext *pMsgContext)
 ^
"desc.c", Warning[Pe177]:
 function "DescOemEnableProgrammingMode" was declared but never referenced
```

When does this happen:

At runtime/compile time.

In which configuration does this happen:

Configurations created in an older delivery affected by ESCAN00061312:
"Not possible to support negative responses while suppressing positive response for service $A5 03"
AND
The 'Reload all description files' button on the 'Diag_CanDesc_Kwp' page in GENy has NOT been pressed at least once since moving from the older delivery.

Resolution Description:

Workaround:

Press the 'Reload all description files' button on the 'Diag_CanDesc_Kwp' page in GENy and then save the configuration. This process only needs to be done once.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00073999	Signal handler names have wrong names after deletion of some signals of a DID
Component@Subcomponent:	Diag_CanDesc__coreBase@GenTool_Geny_CANdesc
First affected version:	6.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

After reloading a modified CDD file, where signals of a DID were deleted, the signal handlers of the DID have the names of the deleted signals.

When does this happen:

After deleting some signals within a signal list of a DID in CANdela Studio and reloading the modified CDD file
within GENy.

In which configuration does this happen:

Signal handlers are used

Resolution Description:

Workaround:

Empty the field "CANdela document name" and click "Reload all description files". The configuration is now empty and no old names are stored. Afterwards import the CDD file again.

OR

For the affected signals change the Signal Handler Type to "Direct Access" and back to "Signal Handler" again

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00078197	Missing first response for service 0xA9 0x81 request
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	6.06.00
Fixed in versions:	6.21.00

Problem Description:

What happens (symptoms):

For a service 0xA9 0x81 request the first response with the first DTC is not sent on the bus.

When does this happen:

Always during run-time in the following scenario:
- A scheduled DPID is currently read out (could take multiple task calls)
- During the reading the service 0xA9 0x81 is requested

In which configuration does this happen:

Service 0xAA is configured
AND
Service 0xA9 0x81 is configured

Resolution Description:

Workaround:

Create a user config file with the following content and add it in GENy to the component CANdesc:

```
#ifdef DESC_UUDTNET_DISABLE_MULTI_CLIENT
# undef DESC_UUDTNET_DISABLE_MULTI_CLIENT
# define DESC_UUDTNET_ENABLE_MULTI_CLIENT
#endif
```

Afterwards generate CANdesc again.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00081145	Validity Bits for Oem GM (Consistency Checks)
Component@Subcomponent:	GenTool_GenyObjectHierarchyCan@GenTool_Geny
First affected version:	1.09.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

GENy in combination with the Il_Vector_Gm does not generate.

When does this happen:

Always and immediately if the configuration has been created with an inconsistent dbc file.

In which configuration does this happen:

Any configuration with validity bits, where
a validity bit is part of a rx message
AND
a signal using a validity bit is mapped to the current ECU
AND
the validity bit itself is NOT mapped to the current ECU.

Resolution Description:

Workaround:

Change your dbc file and map the validity bit signal to the ECU. Update the configuration in GENy or create a new the configuration with the updated dbc file.

Resolution:

The described issue is corrected by modification of all affected work-products.

2.3 Apparent Issues

Apparent issues are detected immediately when using the basic software. If an issue does not show up while working with the basic software the ECU project is not affected by the issue. Apparent issues may or may not have workarounds.

Index

ESCAN00049589	Compile error: direct signal access feature in CANdesc does not consider far memory pointers
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	1.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

Compile error for mismatching pointer type assignment.

When does this happen:

At compile time.

In which configuration does this happen:

- CANdesc
AND
- Direct signal access to RAM/ROM objects is used.
AND
- FAR memory

Some services such as the UDS ones 0x22/0x2A and 0x2E, can be processed on signal level. If they are processed on signal level it is possible to choose "Direct Access" as Signal Handler Type. In this case, CANdesc reads or writes the value of signal direct of/to a variable. (The name of the variable is configured in the cdd file or GENy.) If this variable is located in FAR memory a Compiler/Linker warning or error will occur.

Resolution Description:

Workaround:

Avoid direct signal access to such objects and implement the main-handler within the application code. (Choose "Signal Handler" for the Signal Handler Type and copy the data that is located in the FAR memory in the application callback for this signal.)

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00055957	appdesc.c missing line feed (LF) after carraige return (CR) on some lines
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	5.07.26
Fixed in versions:	

Problem Description:

What happens (symptoms):

The appdesc.c file is missing the line feed (LF) character at the end of certain lines. It should follow the carraige return (CR) character. This will cause compilers and debuggers to display the incorrect line of source code. Additionally, some IDEs will complain that the line feed character is missing.

When does this happen:

At generation time.

In which configuration does this happen:

All configurations.

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00069876	Incorrect Description for Calibration Attribute nmMaxApplShutDownDenyCnt
Component@Subcomponent:	CBD_TechRef_GmlanCalibration@Doc_TechRef
First affected version:	2.01.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The description in chapter 6.7 for calibration attribute nmMaxApplShutDownDenyCnt suggests that this attribute can be modified in the GENy GUI, but the selection doesn't exist. The documentation will be updated to remove this suggestion.

nmMaxApplShutDownDenyCnt can be modified in the generated handler calibrations file gmlcal.c.

When does this happen:

Always

In which configuration does this happen:

All

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00070445	The P2 timings can be changed in the tool GUI but the new values have no effect on CANdesc code generation
Component@Subcomponent:	Diag_CanDesc__coreBase@GenTool_Geny_CANdesc
First affected version:	6.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The user is able to modify the default P2 timings in the GENtool GUI, but the new values are not used during the CANdesc code generation. As a result the default P2 timings are only applicable.

When does this happen:

At CANdesc configuration resp. code generation time.

In which configuration does this happen:

Any KWP2000 configuration that shall use P2 timings other than the default ones.

Resolution Description:

Workaround:

The P2/P2* timings are set automatically according to the GM specification to the values of 75 ms/ 5000 ms. Usually, there should be no need to change the P2/P2* timings in GENy to a different value.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00071069	The description of service 0x12 is out-dated
Component@Subcomponent:	Diag_CanDesc_Oem@Doc_TechRef_Kwp_Gm
First affected version:	3.00.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

The description of service 0x12 doesn't consider the changes made with CANdesc 6.
Only one application callback on SID level is generated instead of two, for each sub-function.

When does this happen:

Always.

In which configuration does this happen:

Any.

Resolution Description:
Workaround:

6.2 Service ReadFailureRecordData ($12)
CANdesc generates only one function callback (main-handler) for all service $12 requests and does not offer any special support for this service. Therefore all dispatching and validation steps (e.g. dispatching on sub-function level, check the request length or validate the PID parameter if applicable), as well as the assembly of the response message (including the sub-function byte) have to be performed by the application.

6.2.1 Service ReadFailureRecordIdentifiers ($12 $01)
Depending on the report type requested (PID or DPID) the application must place one of the following values into the first data byte of the response message:
0x00 - for report by PID
0x01 - for report by DPID

Note
The ECU can support either reports by PID or DPID, but not both.

6.2.2 Service ReadFailureRecordParameters ($12 $02)
CANdesc does not automatically include the PID parameter in the response message for service $12 $02. The application must perform this task.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00076919	"unknown exception occurred" occurs during DBC update
Component@Subcomponent:	Il_Vector@GenTool_Geny
First affected version:	1.05.02
Fixed in versions:	

Problem Description:
What happens (symptoms):

The following error message is displayed:

"[Error] an unknown exception occurred when updating VObjectConsumer: <unnamed>"

Additionally, it is not possible to reload the GENy configuration after saving and closing it. Upon reloading, the following error message is displayed:

"[Error] Unknown exception during ECU load"

When does this happen:

When performing the 'Upgrade Database' action in GENy (the "X->Z" button on the toolbar).

In which configuration does this happen:

When an Rx message in the original DBC file is changed to a Tx message in the new DBC file
Note: This can happen when one identity of a multiple identity module receives its own message from another identity
AND
This message is configured to use a common buffer in GENy
(both conditions must be true for the issue to occur)

Resolution Description:
Workaround:

Perform the following steps in GENy:

1. On the Tx or Rx Messages page, for the affected message, click in the "Common Buffer" field and delete all the text in this box
2. On the Tx or Rx Messages page, change the affected message buffer type from "Common Buffer" to "Own Buffer"
3. Perform the DBC update as usual

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00079945	Os cat2 interrupts for Autosar OS are not supported in CANbedded configurations
Component@Subcomponent:	DrvCan__HllIdxCrx@Doc_TechRef
First affected version:	3.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

It is not possible to select OS cat 2 interrupts for the CAN driver in GENy.

When does this happen:

This happens during configuration in GENy.

In which configuration does this happen:

---

in CANbedded configurations where OS Type is set to "Autosar" and the CAN interrupt has to be set to OS cat 2.

Resolution Description:

Workaround:

The OS Type "OSEK" can be used instead. The application has to provide the file "osek.h" which has to include "os.h".

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00083461	Description missing, that Block Size and STmin always reset when a connection is terminated
Component@Subcomponent:	Tp_Iso15765@Doc_TechRef
First affected version:	2.00.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

When using the APIs TpRxSetBS and TpRxSetSTmin, the values are not persisted after the end of a connection. This means that after the connection is terminated, also the APIs TpRxGetBS and TpRxGetSTmin don't return the value which have been set before.
The APIs need to be called again before the next connection (e.g. from the context of ApplTpGetBuffer).

Although this behavior is intended, it is not described in the TechRef.

When does this happen:

Whenever using the APIs to dynamically change the flow control parameter

In which configuration does this happen:

TP_USE_EXTENDED_API_BS == kTpOn
or
TP_USE_EXTENDED_API_STMIN == kTpOn

Resolution Description:
Workaround:

If FC parameters shall be persisted after a connection is terminated, the application must call TpSetBS / TpRxSetSTmin at the beginning of each connection, ideally from the context of the ApplTpGetBuffer call-out.

If the value of the parameters read back by TpGetBS / TpGetSTmin shall be the same as set before by the according APIs, even if the connection is terminated, it must be persisted in the application.

Resolution:

The described issue is corrected by modification of all affected work-products.

2.4 Not Released Functionality

Not released functionalities are modules and features that have not yet passed a complete development cycle (they are e.g. not or only partly tested). For serial production projects the integrator has to ensure that all BETA features are disabled as indicated. If a ESCAN affects a complete BSW module, the module must not be used for serial production.

No issue to be reported.

2.5 Compiler Warnings

As a service we also provide the known compiler warnings. The occurrence of a compiler warning may depend on the used basic software configuration and compiler settings.

Index

ESCAN00027183	Compiler warning: condition is always true Diag_CanDesc__coreBase@Implementation
ESCAN00027751	Compiler warning for cast to smaller type for "failedByteMask" Diag_CanDesc__coreBase@Implementation
ESCAN00029697	Compiler warning for useless assignment on API DescPmClientCheckPid Diag_CanDesc__coreBase@Implementation
ESCAN00031035	Compiler Warning: variable "timer" in "DescRdpiDeletePid" is possibly uninitialized Diag_CanDesc__coreBase@Implementation
ESCAN00047283	IL flags are declared without the "volatile" keyword. Il_Vector@Implementation
ESCAN00058378	Compiler warning: narrowing or signed-to-unsigned type conversion found Diag_CanDescGgdaExt_Gm@Implementation
ESCAN00059701	Compiler warning: condition is always true" in the IlTxTimerTask, IlTxStateTask and IlTxRepetitionsAreActive Il_Vector@Implementation
ESCAN00079828	Compiler warning: CANdesc Variable "reason" Set But Never Used Diag_CanDesc__coreBase@Implementation
ESCAN00081827	Compiler warning: Truncating assignment in DescUsdtNetIsoTpCopyToCan Diag_CanDesc__coreBase@Implementation
ESCAN00081836	Compiler warning: Truncating assignment in DescConfirmation Diag_CanDesc__coreBase@Implementation
ESCAN00081850	Compiler warning: Truncating assignment in DescICNGetResponseData Diag_CanDesc__coreBase@Implementation
ESCAN00081852	Compiler warning: Truncating assignment in DescUudtNetCANTxReserveResource Diag_CanDesc__coreBase@Implementation
ESCAN00081853	Compiler warning: Truncating assignment in DescPidDispatcher Diag_CanDesc__coreBase@Implementation
ESCAN00081861	Compiler warning: Truncating assignment in DescContextStateTask Diag_CanDesc__coreBase@Implementation
ESCAN00081862	Compiler warning: Truncating assignment in DescUpdateScheduler Diag_CanDesc__coreBase@Implementation
ESCAN00081863	Compiler warning: Truncating assignment Diag_CanDesc__coreBase@Implementation
ESCAN00081864	Compiler warning: Truncating assignment in DescGetDynDpidHandle Diag_CanDesc__coreBase@Implementation
ESCAN00081869	Compiler warning: Truncating assignment in DescDefDynDpidAppendPidDefinition Diag_CanDesc__coreBase@Implementation
ESCAN00083566	Compiler warning: variable g_descSchedulerTimer set but never used Diag_CanDesc__coreBase@Implementation
ESCAN00083588	Compiler warning: variable g_descRdpiStateCtrl is set but never used Diag_CanDesc__coreBase@Implementation
ESCAN00085287	Canbedded only: Compiler warning: Variable 'canPendingTemp' was set but never used DrvCan_Sh2RscanLl@Implementation
ESCAN00086295	Compiler warning: Infinite loop possibility Diag_CanDesc__coreBase@Implementation

Index

ESCAN00027183	Compiler warning: condition is always true
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	5.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The compiler produces the warning "variable ondition is always true" when compiling desc.c.

When does this happen:

At compile time

In which configuration does this happen:

Tasking compiler is used

Resolution Description:

Workaround:

No workaround available.

Resolution:

This does not affect the behavior of the compiled code, and can be safely ignored.

ESCAN00027751	Compiler warning for cast to smaller type for "failedByteMask"
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	3.01.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

Compiler warning message for the assignment:

*failedByteMask = (vuint8)(0x02 << *failedByteMask);

But there is no real danger of losing information by casting down to a smaller type since the code generator does not allow more than 7 (seven) sub-service bytes in the request message. So skipping the SID (bit 0) does not lead to losing the MSB and the value of the failedByteMask cannot be greater than six.

When does this happen:

At compile time.

In which configuration does this happen:

-CANdesc/CANdescBasic

Resolution Description:
Workaround:

Ignore the warning

Resolution:

This ESCAN will not be resolved, since the fix might require more resources on the ECU. The code generator assures that there will be no overflow on the shift operation.

ESCAN00029697	Compiler warning for useless assignment on API DescPmClientCheckPid
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	4.02.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

The compiler generates a warning message for useless assignment (a = a;) .

When does this happen:

At compile time.

In which configuration does this happen:

- CANdesc
AND
- Service 0x22 is supported with multiple DIDs in single request.
AND
- Service 0x2C is not supported.

Resolution Description:
Workaround:

Ignore this warning since it is only because of an useless assignment.

Resolution:

The described issue will not be corrected, since the solution will require more ROM resources.

ESCAN00031035	Compiler Warning: variable "timer" in "DescRdpiDeletePid" is possibly uninitialized
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	5.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The compiler produces the warning "variable 'timer' is possibly uninitialized" when compiling desc.c.

When does this happen:

At compile time

In which configuration does this happen:

If service 0xAA supported and at least one periodic transmission mode is supported (i.e. not only "send one response" is supported).

Resolution Description:

Workaround:

No workaround available.

Resolution:

This warning is incorrect and can be safely ignored.

ESCAN00047283	IL flags are declared without the "volatile" keyword.
Component@Subcomponent:	Il_Vector@Implementation
First affected version:	3.10.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

IL flags (Indication, FirstValue, Confirmation, Timeout) are declared without the "volatile" keyword. Read and Write access to IL flags has no effect due to a Read-Modify-Write problematic.

e.g.
FlagA and FlagB are in the same byte and set on interrupt level

this sequence is executed on task level:
disable int;
clear FlagA; /*1*/
enable int;
... /*3*/
disable int;
clear FlagB; /*2*/
enable int;

The compiler might optimize this sequence and the flag read and write ALWAYS fails:

read the byte at 1), modify the local copy and write the byte at 2)
if the byte is written on interrupt level at 3), the data is lost.

When does this happen:

At runtime (This Problem has been found by a review and has never been detected in a ECU)

In which configuration does this happen:

- This issue highly depends on the used compiler and compiler options.
- Preemptive IL flag access is used (e.g. interrupt system)

Resolution Description:

Workaround:

Review the optimization configuration of your compiler.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00058378	Compiler warning: narrowing or signed-to-unsigned type conversion found
Component@Subcomponent:	Diag_CanDescGgdaExt_Gm@Implementation
First affected version:	2.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The following compiler warning occurs:

warning (dcc:1643): narrowing or signed-to-unsigned type conversion found: unsigned int to unsigned short

This warning occurs for the following code in GgdaRdiRxTask:

```
 /* send the DTC number and the FTB */
 vuint16 DTCNr = ((vuint16) ggdaContexts[context].uudtPrimBuffer[1] << 8)
 | (vuint16) ggdaContexts[context].uudtPrimBuffer[2];
```

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

Configurations which use service $A9.

Resolution Description:

Workaround:

The warning can be safely ignored.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00059701	Compiler warning: condition is always true" in the IlTxTimerTask, IlTxStateTask and IlTxRepetitionsAreActive
Component@Subcomponent:	Il_Vector@Implementation
First affected version:	2.42.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

Compiler warns for "condition is always true" in the IlTxTimerTask, IlTxStateTask and IlTxRepetitionsAreActive API. This may happen depending on the configuration.

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

IlTxTimerTask, IlTxStateTask: Any configuration with exactly one tx message.
IlTxRepetitionsAreActive: Any configuration with exactly one tx message and the API is configured. (IL_ENABLE_SYS_TX_REPETITIONS_ARE_ACTIVE_FCT must be defined)

Hint:

The compiler warning is known and has been analyzed thoroughly for its impact on the code. Nevertheless it will not be fixed due to the rare configuration. The code uses a while loop with a counter and can probably replaced by a for loop, but other compilers or codeanalysers may warn about a useless loop. The code exists for about 15 Years and will not be changed.

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00079828	Compiler warning: CANdesc Variable "reason" Set But Never Used
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	6.16.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

A compiler warning is issued for "desc.c, line _____:warning #550: variable "reason" was set but never used". This is found in function DescDefDynDpidAppendPidDefinition.

When does this happen:

Always during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

Service 0x2C is configured

Resolution Description:
Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00081827	Compiler warning: Truncating assignment in DescUsdtNetIsoTpCopyToCan
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	5.07.24
Fixed in versions:	

Problem Description:

What happens (symptoms):

The compiler warning "Truncating assignment" is issued in the function DescUsdtNetIsoTpCopyToCan:

...

vuint8_least i = infoStruct->Length;

...

No issues will result from this warning, because the value passed from the ISO TP will never exceed 255.

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

The data type vuint8_least is mapped to vuint8 (unsigned char) (which is mostly done for 8Bit Controller)
AND
ISO TP from Vector is used

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00081836	Compiler warning: Truncating assignment in DescConfirmation
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	3.01.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The compiler warning "Truncating assignment" is issued in the function DescConfirmation in the usage of the conditional operator:

...

result = (status != kTpSuccess) ? kDescUsdtNetworkAbort:kDescUsdtNetworkOk;

...

No issues will result from this warning, because the two possible values used in the conditional operator have the same type as the variable they are assigned to.

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

ISO TP from Vector is used

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00081850	Compiler warning: Truncating assignment in DescICNGetResponseData
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	3.02.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The compiler warning "Truncating assignment" is issued in the function DescICNGetResponseData in the usage of the conditional operator:

```
...
 copyLen = ((((vuint16)
(g_descUsdtNetInfoPoolDescICN[DICN_CHANNEL_PARAM_VALUE].dataLength -
g_descIcnState[DICN_CHANNEL_PARAM_VALUE].rdIdx)) / kDescICNTempBufferLen) != 0)?
 kDescICNTempBufferLen:
 (g_descUsdtNetInfoPoolDescICN[DICN_CHANNEL_PARAM_VALUE].dataLength %
kDescICNTempBufferLen);
...
```

No issues will result from this warning, because the two possible values which can be assigned are always smaller than 8 byte.

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

ISO TP from Vector is used

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00081852	Compiler warning: Truncating assignment in DescUudtNetCANTxReserveResource
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	6.16.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The compiler warning "Truncating assignment" is issued in the function DescUudtNetCANTxReserveResource for the following assignment:
...
 g_descUudtNetResMgrCtxt.size =
DESC_CHNL_INFO_MSG_BASE_IDX_NXT(CHANNEL_ITER_VALUE) -
g_descUudtNetResMgrCtxt.baseIdx;
...

No issues will result from this warning, because the two operands for the subtraction are always smaller than 255 and the first operand always greater or equal than the second operand.

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

Service 0x2A is configured
AND
UUDT messages for the periodic responses are configured
AND
UUDT messages on more than one channel are configured

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00081853	Compiler warning: Truncating assignment in DescPidDispatcher
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	5.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The compiler warning "Truncating assignment" is issued in the function DescPidDispatcher for the following assignment:

```
...
 iter = g_descMsgContext[DESC_CONTEXT_PARAM_VALUE].reqDataLen;
...
```

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

---

data type vuint8_least is mapped to vuint8 (unsigned char)(mostly done for 8 bit controller)
AND
Service 0x22 is used
AND
Multiple PIDs are allowed to be requested in one request

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00081861	Compiler warning: Truncating assignment in DescContextStateTask
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	4.00.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

The compiler warning "Truncating assignment" is issued in the function DescContextStateTask for the following assignment:
...
g_descInterruptContextCtrl[DESC_CONTEXT_PARAM_VALUE].infoPoolPtr->resType =
(g_descNegResCode[DESC_CONTEXT_PARAM_VALUE] == kDescNrcNone)?
kDescUsdtResponsePositive:kDescUsdtResponseNegative;
...

No issues will result from this warning, because the two possible values used in the conditional operator have the same type as the variable they are assigned to.

When does this happen:

The warning is issued by the compiler during compilation of the code.

In which configuration does this happen:

In all configurations

Resolution Description:
Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00081862	Compiler warning: Truncating assignment in DescUpdateScheduler
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	4.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The compiler warning "Truncating assignment" is issued in the function DescUpdateScheduler for the following assignment:

...

i = pMsgContext->reqDataLen;

...

No issue will result from this warning because the reqDataLen parameter is checked against the constant kDescNumOfPeriodicPids and this constant will be limited by the CANdesc generator to 255.

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

---

data type vuint8_least is mapped to vuint8 (unsigned char)(mostly done for 8 bit controller)
AND
Service 0xAA is used

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00081863	Compiler warning: Truncating assignment
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	4.02.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The compiler warning "Truncating assignment" is issued in...

1) ... the function DescReadDpidStop for the assignment

```
 ...
 i = pMsgContext->reqDataLen;
 ...
```

2) ... the function DescReadDpidOnce for the assignment

```
 ...
 i = pMsgContext->reqDataLen;
 ...
```

An issue would only arise for warnings 1) and 2) if one request contains more than 255 DPIDs. However, the range of allowed DPID values is smaller than 240 elements. Therefore, to request more than 255 DPIDs implies that one DPIDs is requested multiple times in the same request, which is not a use case at all.

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

---

data type vuint8_least is mapped to vuint8 (unsigned char)(mostly done for 8 bit controller)
AND
Service 0xAA is used

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00081864	Compiler warning: Truncating assignment in DescGetDynDpidHandle
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	4.02.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The compiler warning "Truncating assignment" is issued in the function DescGetDynDpidHandle for the following statement:

```
...
 return (g_descDynDpid2GlobalDpidHandle[iter] != globalDpidHandle)?
 kDescNumDynDefinedDpids:iter;
...
```

No issue will arise from that warning because the value range of DPIDs is by definition smaller than 255 .

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

Dynamic DPIDs are supported

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00081869	Compiler warning: Truncating assignment in DescDefDynDpidAppendPidDefinition
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	4.02.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

The compiler warning "Truncating assignment" is issued in the function DescDefDynDpidAppendPidDefinition for the following statement:

```
...
g_descDynDpidTempInfoTable.resDataLength += DescPmGetPidResponseLen(srcPidHandle);
...
```

No issue will arise from that warning because the amount of response data will never exceed 255.

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

Dynamic DPIDs are supported

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00083566	Compiler warning: variable g_descSchedulerTimer set but never used
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	5.00.02
Fixed in versions:	

Problem Description:

What happens (symptoms):

Compiler warns that the variable g_descSchedulerTimer is set but never used.

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

Service 0xAA is configured only with the sub-function 0x01 to read a DPID once.

Resolution Description:

Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00083588	Compiler warning: variable g_descRdpiStateCtrl is set but never used
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	5.04.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

Compiler warns that the variable g_descRdpiStateCtrl is set but never used.

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

Service 0xAA is configured only with the sub-function 0x01 to read a DPID once.

Resolution Description:
Workaround:

No workaround available.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00085287	Canbedded only: Compiler warning: Variable 'canPendingTemp' was set but never used
Component@Subcomponent:	DrvCan_Sh2RscanLl@Implementation
First affected version:	3.01.00
Fixed in versions:	4.01.00, 3.13.04

Problem Description:

What happens (symptoms):

The compiler issues a warning like following: Variable 'canPendingTemp' was set but never used.

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

C_ENABLE_CANCEL_IN_HW is defined
and
C_ENABLE_TX_OBSERVE is NOT defined
and
C_ENABLE_CAN_TX_CONF_FCT is NOT defined
and
C_ENABLE_RETRANSMIT is NOT used in the project

Resolution Description:

Workaround:

The compiler warning does not indicate any unsuspected behavior and can be ignored.

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00086295	Compiler warning: Infinite loop possibility
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	5.07.24
Fixed in versions:	6.21.00

Problem Description:

What happens (symptoms):

Compiler warning due to the possibility of infinite loop when the following code is executed

```
while(i--)
 {
 infoStruct->pDestination[i] = infoStruct->pSource[i];
 }
```

There is no possibility of going through an infinite loop in this situation. Ultimately i reaches zero. If i is zero in the first place, while loop will not be executed.

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

---

all Configurations

Hint:

Resolution Description:

Workaround:

Resolution:

ESCAN00088724	Compiler warning: dummy function has no prototype
Component@Subcomponent:	Tp_Iso15765@GenTool_Geny
First affected version:	2.00.00
Fixed in versions:	

Problem Description:

What happens (symptoms):

Compiler warns for a function definition without prototype in tp_par.c

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

TP Class = Static Normal Multip TP

AND

at least one connection group does not use the same callbacks as the other connection groups (this also happens automatically if there are unidirectional connection groups)

Hint:

The compiler warning is known and has been analyzed thoroughly for its impact on the code. Nevertheless it will not be fixed because there is a workaround

Resolution Description:

Workaround:

---

1) Provide the missing prototypes with a user config file

2) If possible, disable the coding rule checks of the compiler
For Metrowerks, there is separate "Require Function Prototypes" setting to enable / disable the check

Resolution:

The described issue is corrected by modification of all affected work-products.

ESCAN00089512	Compiler warning: braces around scalar initializer [enabled by default]
Component@Subcomponent:	Diag_CanDesc__coreBase@Implementation
First affected version:	3.02.00
Fixed in versions:	

Problem Description:
What happens (symptoms):

When the size of the array g_desc19UsedExtDatRecIds is reduced to only one element as follows:

```
V_MEMROM0 static V_MEMROM1 vuint8 V_MEMROM2 g_desc19UsedExtDatRecIds[1] =
{
{ 0x01 }
};
```

the compiler error: 1510:3: warning: braces around scalar initializer [enabled by default] is issued.

When does this happen:

The warning is issued by the compiler during compilation of the code in case the configuration is as described below.

In which configuration does this happen:

When only one element exists in the array g_desc19UsedExtDatRecIds.

Resolution Description:
Workaround:

Ignore the warning.

Resolution:

The described issue is corrected by modification of all affected work-products.

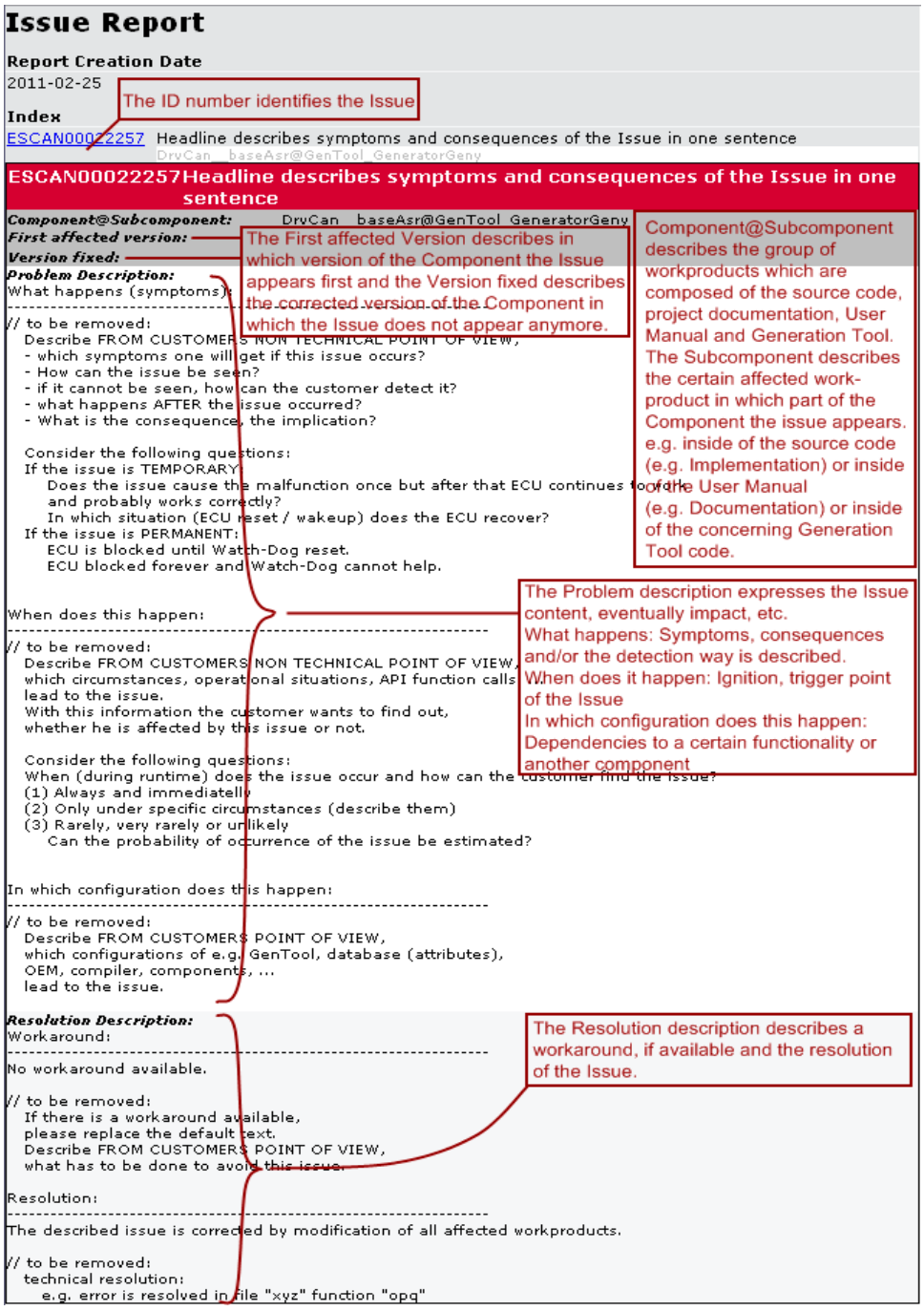
3. New Issues for Information

Issues which should not have an effect on the usage of the license as the issues are relevant for use cases other than those defined in the questionnaire. The list contains issues that have been detected since the last report.

Issues listed in this section are not relevant for the use case that has been documented in the questionnaire provided to Vector. However, the issues may be relevant for other use cases. Also issues that have been accepted or are tolerated by the OEM (as defined in the questionnaire) are reported here.

No issue to be reported.

4. Report Legend



5. Quality Management Contact

Quality Management
Productline Embedded Software (PES)

Vector Informatik GmbH
Ingersheimer Str. 24
D-70499 Stuttgart

Phone: +49 711 80670-3700
Fax: +49 711 80670-399
eMail: QualityPES@vector.com