Microsoft

winhec

Workshop Taipei 2016

Any questions please contact winhectpe@microsoft.com

Microsoft

Case Study in Porting win32 Libraries

Jia-Jium Lee
Software Engineer

Partner Enablement Team

Agenda

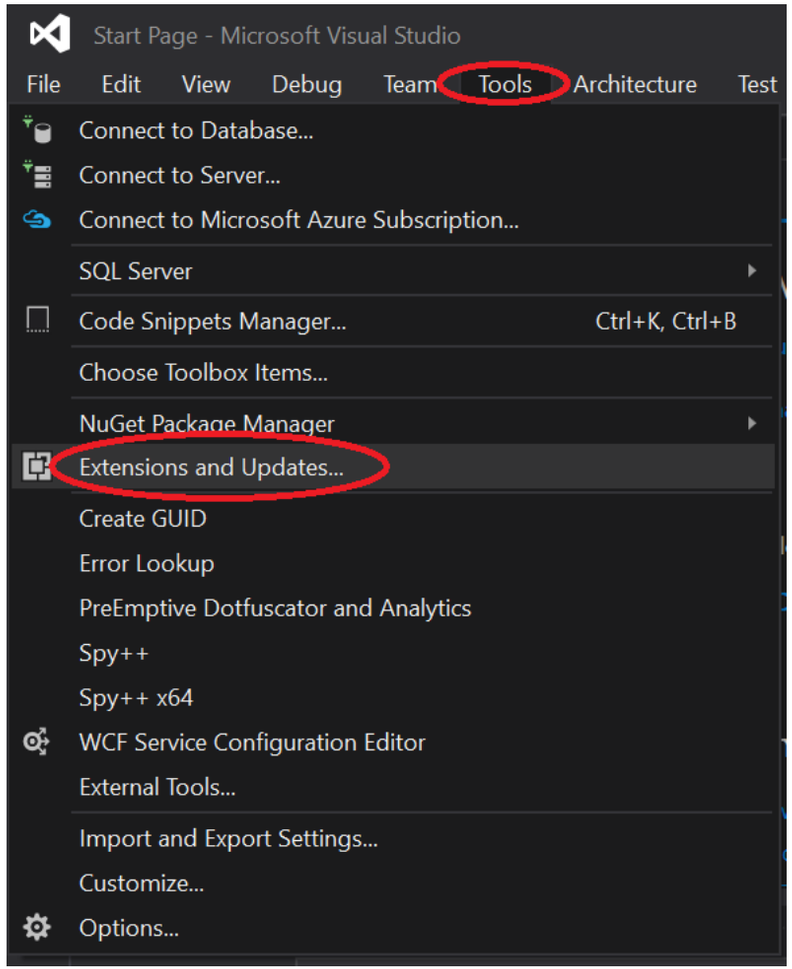
- Development Environment Setup
- Understand Windows 10 IoT Core VC++ Build Environment
- Windows 10 IoT Core Porting Utility
- Case Studies

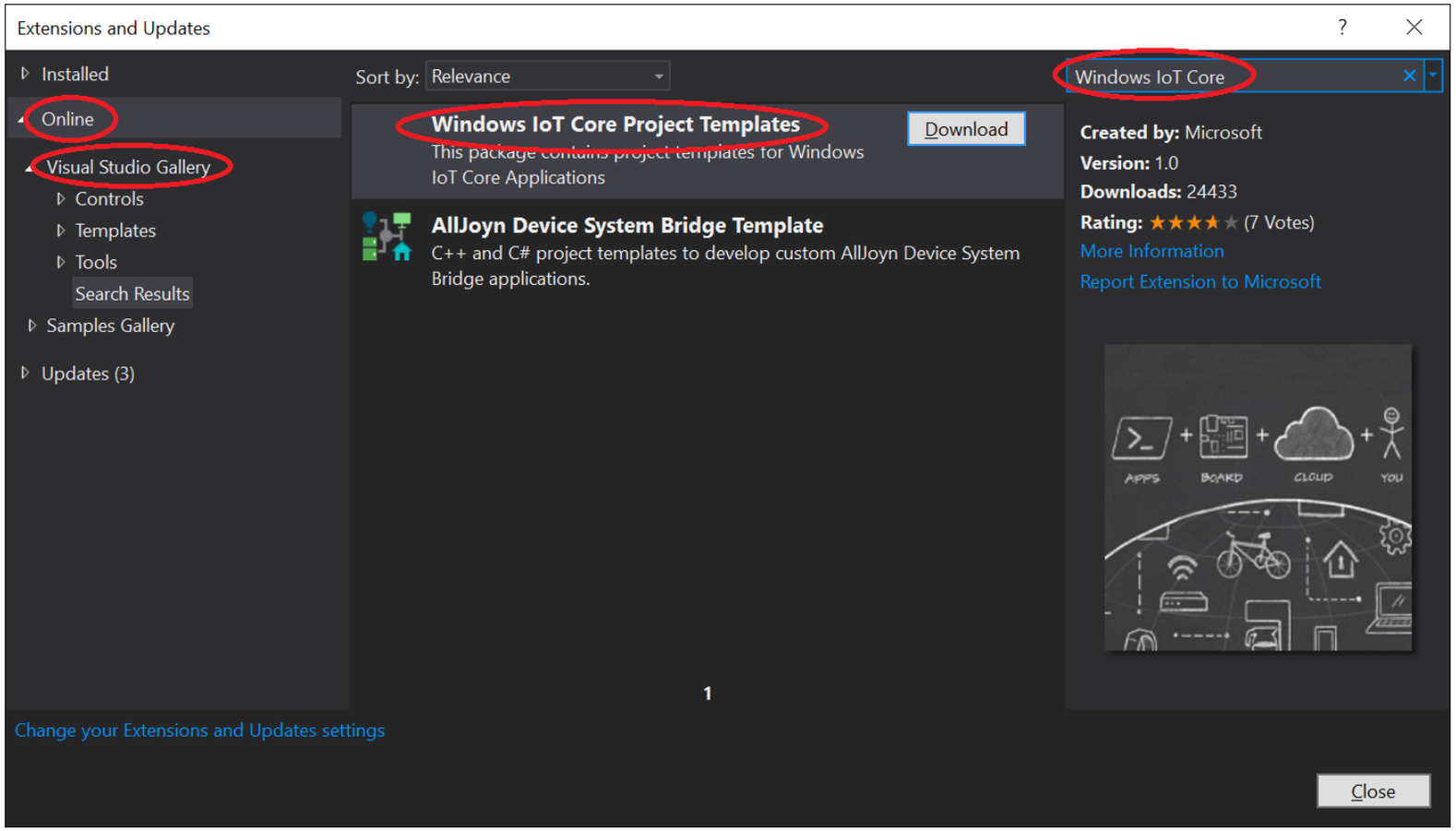
Development Environment Setup

Install Visual Studio 2015 Features

- If you try to Install Update 1
  - Update to Windows 10 November Update first
- Select following features
  - Programming Languages
    - ✓ Visual C++
  - Universal Windows App
  - Common Tools
    - ✓ Git for Windows (Updated)
    - ✓ GitHub Extension for Visual Studio
    - ✓ Visual Studio Extensibility Tools

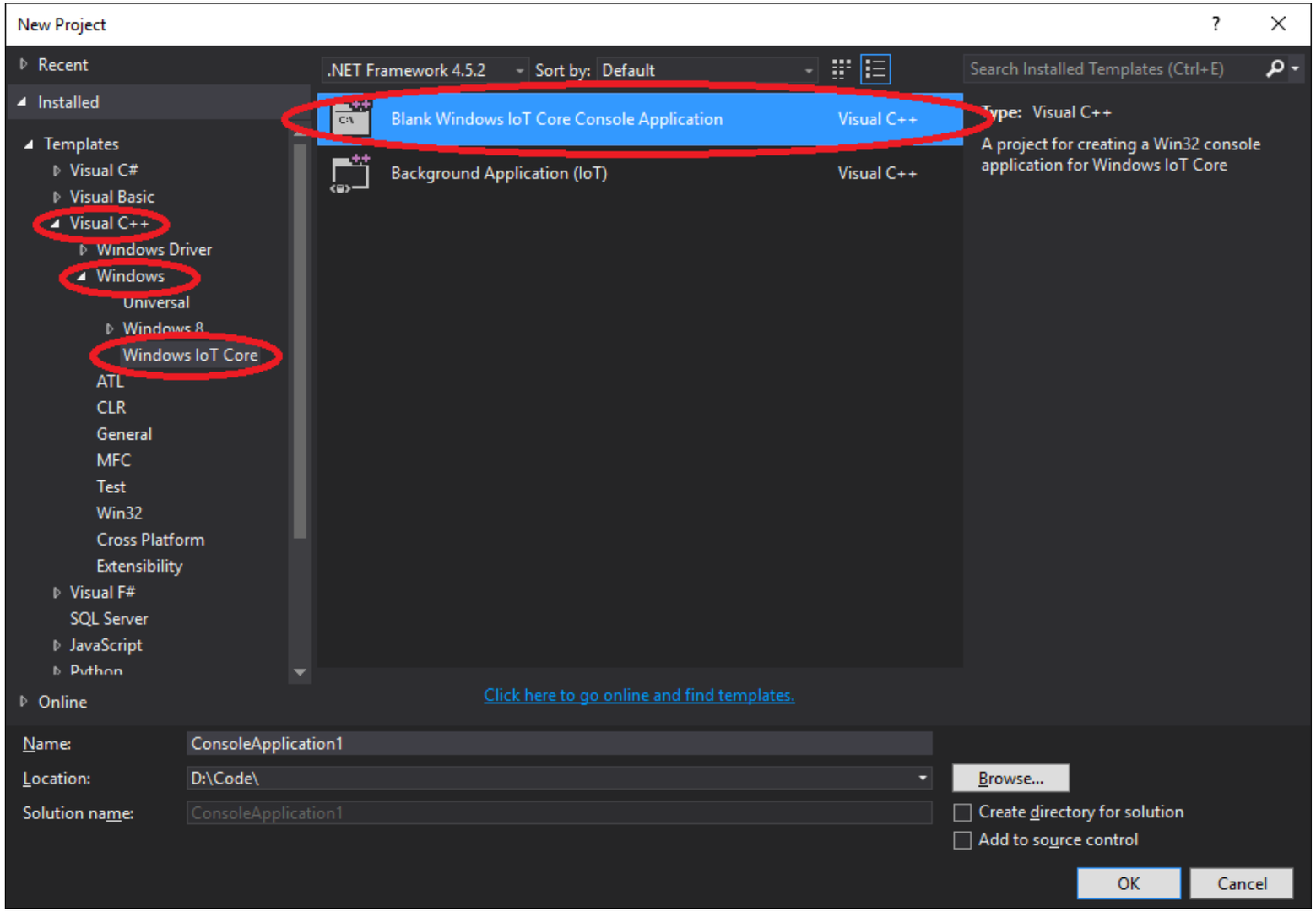
Download Windows 10 IoT Core Project Template





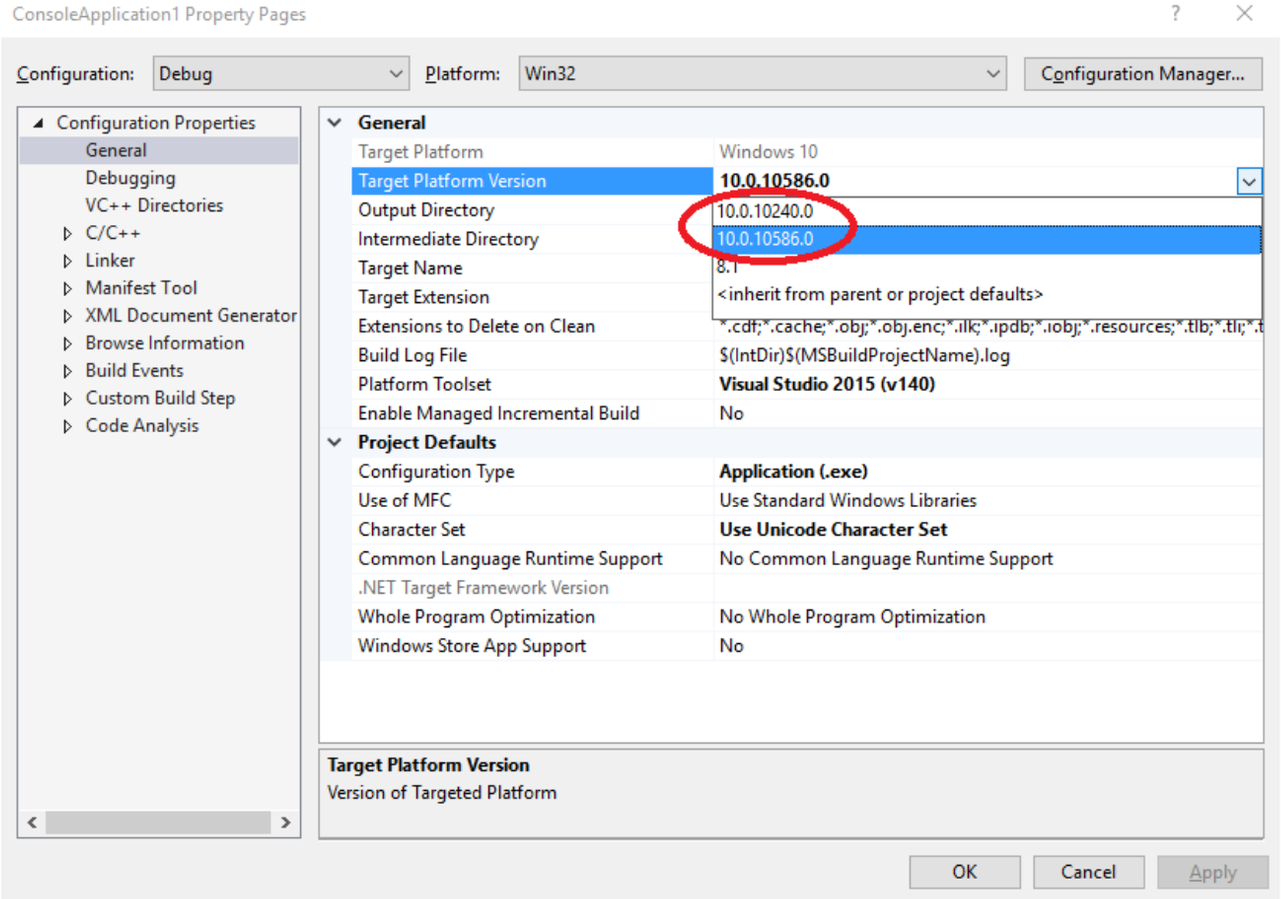
Understand Windows 10 IoT Core VC++ Build Environment

Windows IoT Core Console App Project



Target Platform

- Windows 10 IoT Core only supports 32 bits (x86 ARM)
- Target Platform Version - 10586



Check Include/Library Path and linked Libraries

- Project from default template
  - Should be fine
- Project from exiting or old project files
  - Check include path and library path
  - Check whether link to other Win32 libraries
  - Ensure link to onecoreuap.lib

Windows 10 IoT Core Porting Utility

IoTAPIPortingTool

- Checking Win32 APIs Linking Problem
  - must be run from a Visual Studio Developer Command Prompt
  - IoTAPIPortingTool.exe <path> [-os]
- http://ms-iot.github.io/content/en-US/win10/tools/IoTAPIPortingTool.htm
  - Git Clone
  - Pull the latest source
  - Build

IoTAPIPortingTool Sample Result

- DLL NAME api-ms-win-*
  - should be OK
- DLL NAME vcruntime*
  - Need VC runtime

	DLL NAME	FUNCTION NAME
Debug\ConsoleApplication1.exe	kernel32.dll	GlobalMemoryStatus
Debug\ConsoleApplication1.exe	api-ms-win-core-debug-l1-1-1.dll	IsDebuggerPresent
Debug\ConsoleApplication1.exe	api-ms-win-core-errorhandling-l1-1-1.dll	RaiseException
Debug\ConsoleApplication1.exe	api-ms-win-core-errorhandling-l1-1-1.dll	UnhandledExceptionFilter
Debug\ConsoleApplication1.exe	api-ms-win-core-errorhandling-l1-1-1.dll	SetUnhandledExceptionFilter
Debug\ConsoleApplication1.exe	api-ms-win-core-errorhandling-l1-1-1.dll	GetLastError
Debug\ConsoleApplication1.exe	api-ms-win-core-processthreads-l1-1-2.dll	GetCurrentProcess
Debug\ConsoleApplication1.exe	api-ms-win-core-processthreads-l1-1-2.dll	GetCurrentProcessId

Demo

Summary

- Only link to onecoreuap.lib + your lib
- Check Additional Dependencies
- Check #pragma comment(lib, “????”)
  - Executable will be linked to library ????

Cases Study

Case Study – OpenSSL (1)

- Functionalities
  - Implement TLS and SSL protocols
  - General purpose cryptography library
  - Please refer to https://www.openssl.org/
- Source File Problems
  - crypto\rand\rand_win.c - Function RAND_poll()
    - ✓This function generate random numbers from memory usage
    - ✓Unresolved API – GetMemoryStatus
    - ✓Fixed by – GetMemoryStatusEx under #ifdef WIN_IOT
    - ✓Or a trusted random number generator

Case Study – OpenSSL (2)

- Source File Problems
  - crypto\rand\rand_win.c - readscreen()
    - ✓This function generate random numbers from screen
    - ✓Unresolved GDI APIs – CreateCompatibleDC, CreateCompatibleBitmap, …
    - ✓Fixed by – rand_s() under #ifdef WIN_IOT
    - ✓Or a trusted random number generator
  - crypto\cryptlib.c - Function OPEN_isservice()
    - ✓Checks the executable is a service or not
    - ✓Unresolved APIs – GetDesktopWindow, GetProcessWindowStation, GetUserInformationObject
    - ✓Fixed by
      - Call into callback function only under #idef WIN_IOT
      - Provide below callback in your executable

Case Study – OpenSSL (3)

```
#ifdef __cplusplus
extern "C" {
#endif
// return 1 for a service or return 0 for not a service
extern  __declspec(dllexport)  Int _OPENSSL_isservice() { return 1; }
#ifdef __cplusplus
}
#endif
```

Case Study – OpenSSL (4)

- Source File Problems
  - ➢crypto\cryptlib.c - Function OPENSSL_showfatal()
    - ✓Show error message
    - ✓Unresolved APIs – RegisterEventSource, ReportEvent, DeregisterEventSource, MessageBox
    - ✓Fixed by
      - Don’t show error message
      - Partner don’t need event log functionality here
      - Apply a stub winsysstub.dll to provide above APIs without meaningful implementation

Case Study – OpenSSL (5)

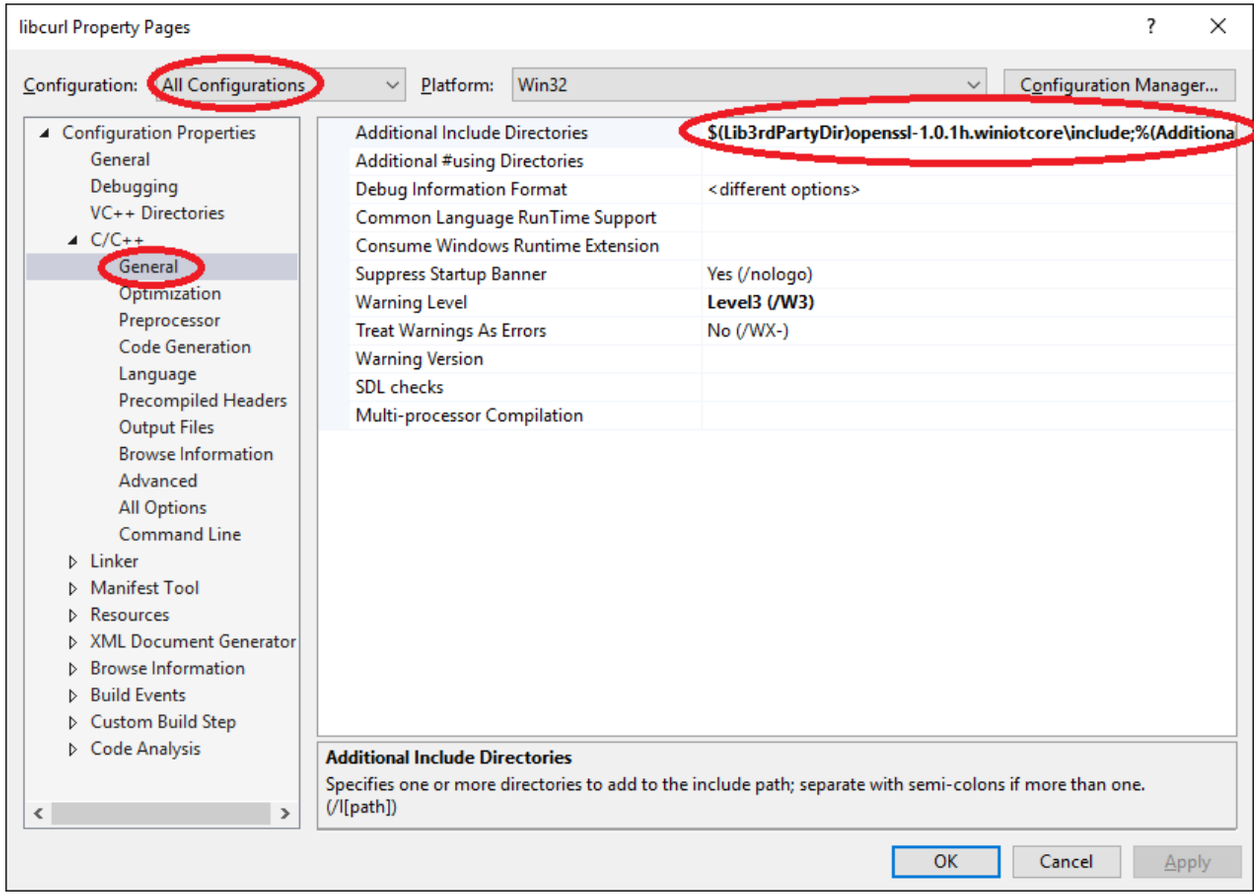
- Build Problems
  - ➢ms\ntdll.mak (or nt.mak)
    - ✓Link to ws2_32.lib gdi32.lib advapi32.lib crypt32.lib user32.lib (IoT only support onecoreuap.lib)
      - Link to onecoreuap.lib winsysstub.lib
      - Please build winsysstub project only in IDE first
      - Add libpath to winsysstub
    - ✓We need to define WIN_IOT macro when compile
    - ✓Summary in file INSTALL.WIOTCORE

Case Study – Curl (1)

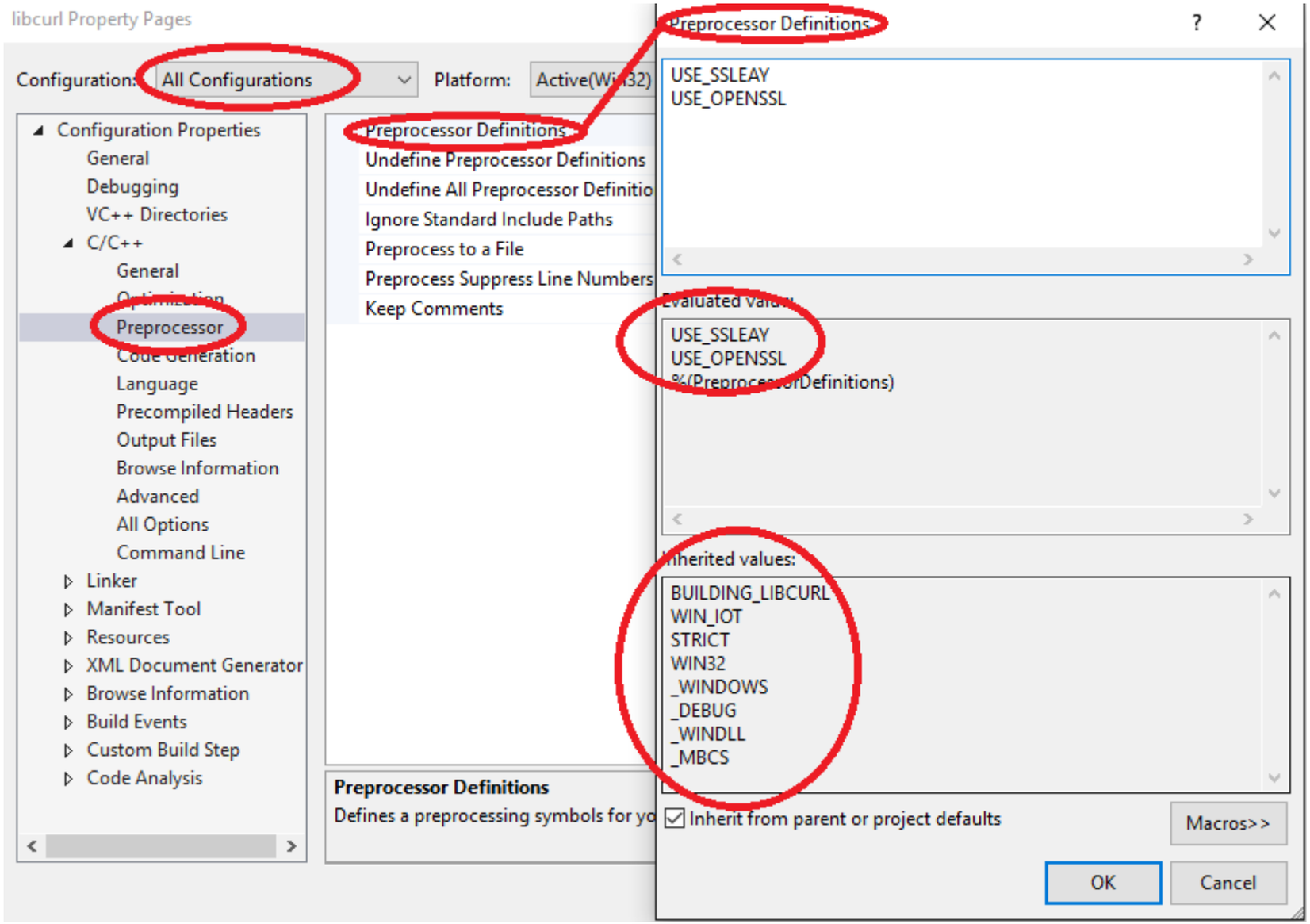
- Funcationalities
  - Transfereing data with URL syntax
  - Support DICT, FILE, FTP, FTPS, Gopher, HTTP, HTTPS, IMAP, IMAPS, …
  - Please refer to http://curl.haxx.se/
- Source File Problems
  - lib\connect.c - Function Curl_sndbufset()
    - ✓Check version
    - ✓Unresolved API – VerifyVersionInfo
    - ✓Fixed by - GetVersionEx

Case Study – Curl (2)

- Build Problems – for **simple configuration** (Debug/Release) only

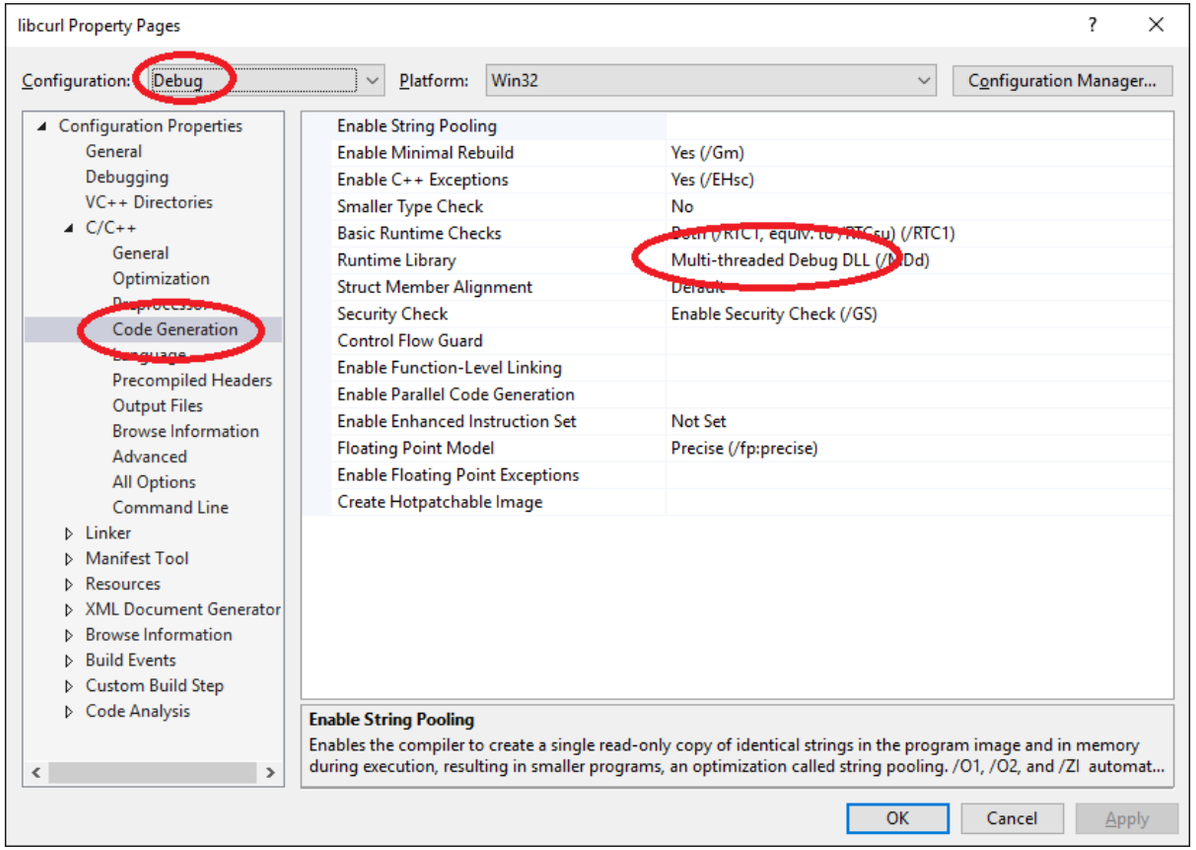


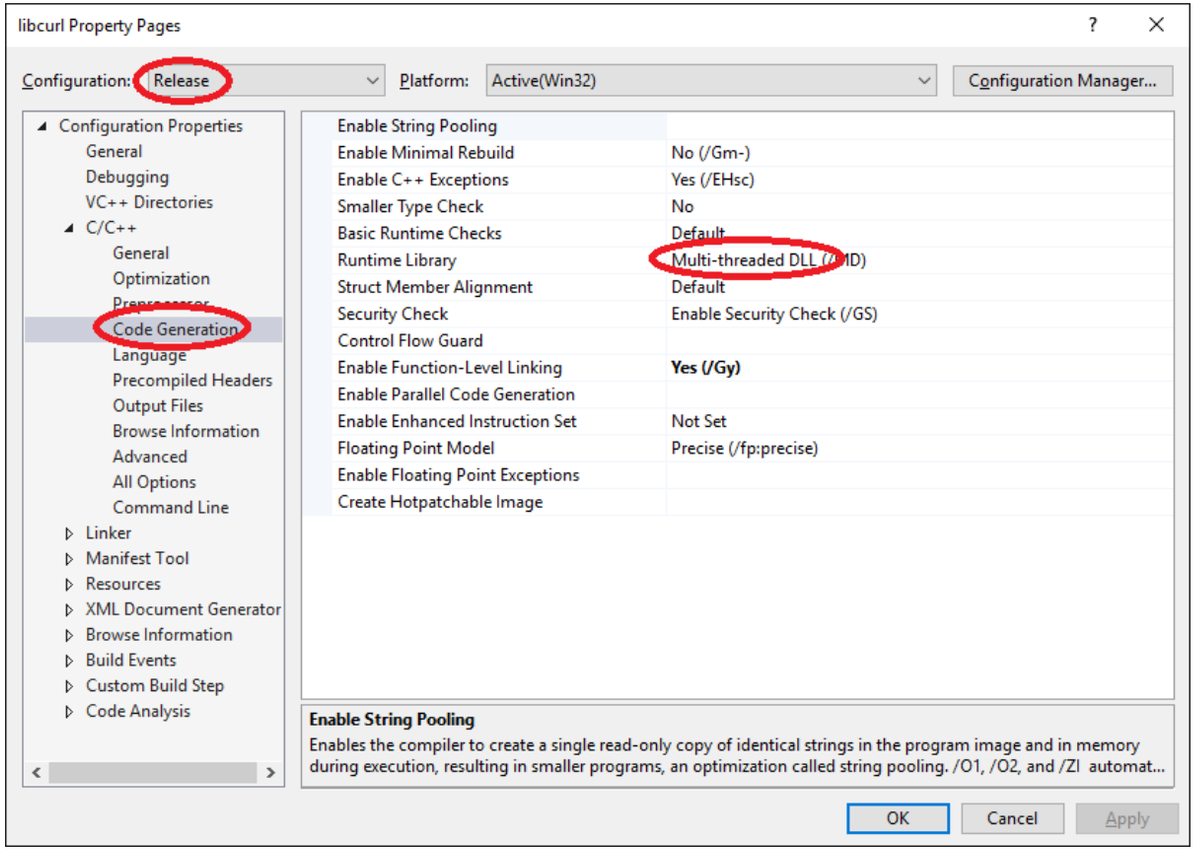
Case Study – Curl (3)



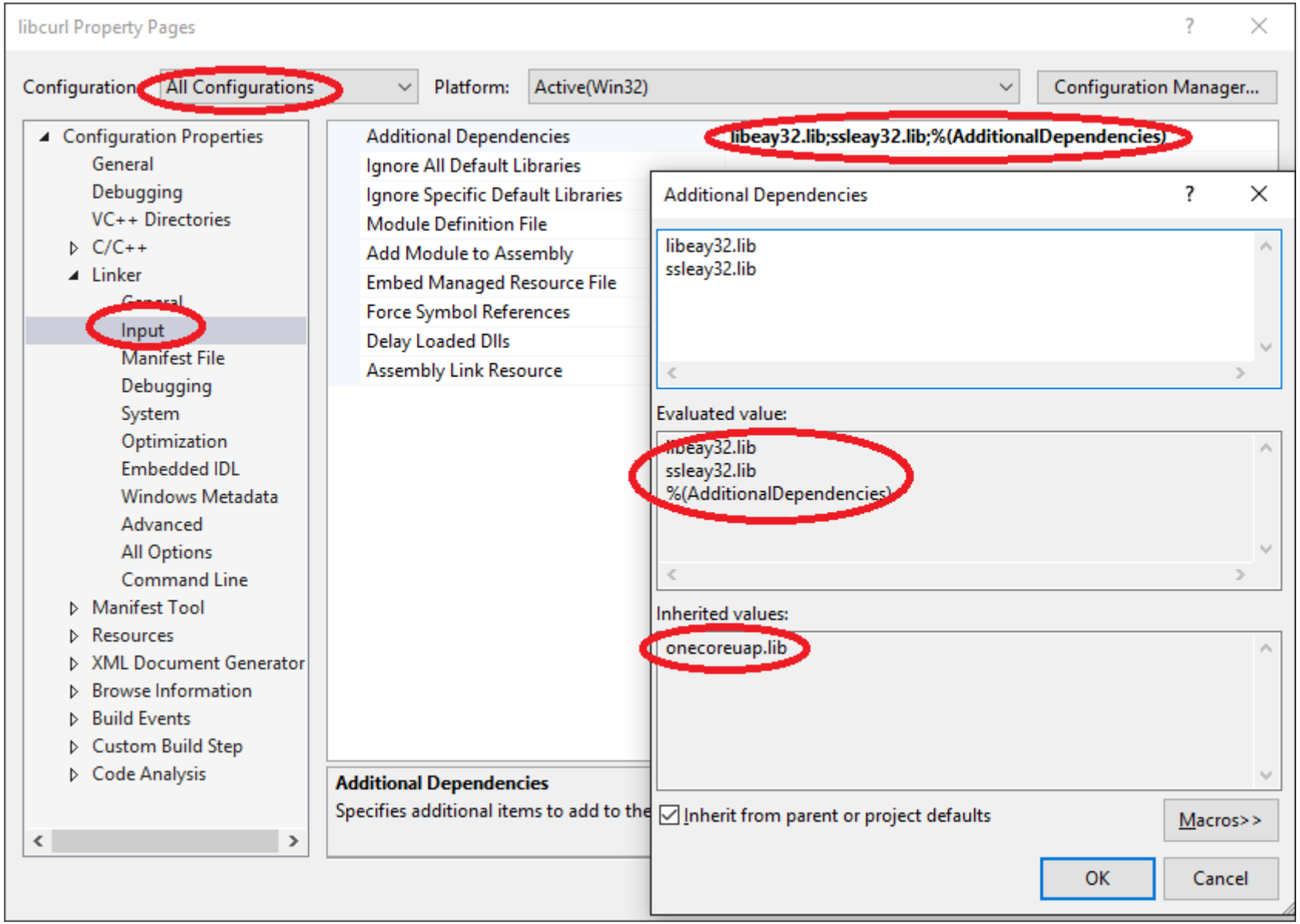
Case Study – Curl (4)

- Configuration: Debug (Multi-threaded Debug DLL)
- Configuration: Release (Multi-threaded DLL)





Case Study – Curl (5)



Hint for Complex Configurations

- Warning
  - ➢If you are going to modify the .vcxproj file directly, backup it first!
  - ➢The tabs and spaces might be different in .sln file!
- Complex 40 configurations
  - ➢DLL Debug – DLL OpenSSL – DLL LibSSH2|Win32 …
  - ➢Check with the git version marked as 10/21/2015 10:09:18 AM
    - ✓Compare with the latest VC14.iotcore\lib\libcurl.vcxproj
  - ➢Might not feasible of checking all property settings by Properties
    - ✓Check .vcxproj is easier (?), in case if you are familiar with it

Modify Project File

- Define property elements in libcurl.vcxproj
  - ➢define a property element
    - ✓Just like define a variable in a normal programming language
    - ✓For example, I would like to define a ConfigurationString

```
<!-- Debug == DLL Debug - DLL OpenSSL -->
<ConfigurationString Condition="'$(Configuration)'=='Debug'">Dll Debug – DLL OpenSSL</ConfigurationString>
```

    - ✓Define a ConfigurationReleaseMode to check it is a Debug or Release build

```
<ConfigurationReleaseMode Condition="$([System.String]::new('$(ConfigurationString)').Contains(' Debug'))">Debug</ConfigurationReleaseMode>
<ConfigurationReleaseMode Condition="$([System.String]::new('$(ConfigurationString)').Contains(' Release'))">Release</ConfigurationReleaseMode>
```

    - ✓Use Debug or Release runtime library based on ConfigurationReleaseNode

```
<UseDebugLibraries Condition="'$(ConfigurationReleaseMode)'=='Debug'">true</UseDebugLibraries>
```

With UseDebugLibraries, we don't need to switch between "Multi-threaded DLL" and "Multi-threaded Debug DLL" Runtime Library, We can leave it as default.

Case Study – Libxml2 (1)

- Functionalities
  - XML C parser
  - Please refer to http://www.xmlsoft.org/
- Source File Problems
  - include\win32config.h – snprintf is defined in VC++ 2015 stdio.h

```
#define snprintf _snprintf
#if _MSC_VER < 1900
#define snprintf _snprintf
#endif
```

Case Study – Libxml2 (2)

- Build Problems
  - ➢win32\Makefile.msvc
    - ✓Init LIB – Link to onecoreuap.lib
    - ✓Link to wsock32.lib ws2_32.lib (Removed)
    - ✓Link to kernel32.lib (Removed)
    - ✓Link option /OPT:NOWIN98 (Removed)
    - ✓Summary in file win32\Readme.WinIoTCore.txt

Case Study - Pthreads Win32

- Functionalities
  - POSIX standard threads related APIs
  - Please refer to https://www.sourceware.org/pthreads-win32/
- Souce File Problems
  - pthreads.2\config.h – timespec defined VC++ 2015, GetProcessAffinityMask is not in IoT Core
    - ✓ #undef HAVE_STRUCT_TIMESPEC
    - ✓ #define HAVE_STRUCT_TIMESPEC
    - ✓ #undef NEED_PROCESS_AFFINITY_MASK
    - ✓ #define NEED_PROCESS_AFFINITY_MASK

Case Study – mosquito (1)

- Functionalities
  - Implements MQTT
  - Please refer to http://mosquitto.org/
- Source File Prolems
  - lib\net_mosq.c - Function _mosquitto_socketpair()
    - ✓ Need a CONST IN6_ADDR INADDR_LOOPBACK
    - ✓ Declaration is in ws2ipdef.h, definition is in ws2_32.lib or ws2_32.dll
    - ✓ Create a winstaticstub.lib
      - Under WISEAgent\Library3rdParty\winstaticstub.winiotcore
      - A static library
      - Build from IDE

Case Study – mosquito (2)

- Source File Prolems
  - config.h – snprintf is defined in VC++ 2015 stdio.h
    - ✓ #define snprintf sprintf_s
    - ✓ #if _MSC_VER < 1900
      #define snprintf sprintf_s
      #endif
  - Build from IDE

Name mangling

- For .cpp
  - C++ support function overloading
  - Technique to support function overloading
    - ✓ Not required but implemented by most of vendors
  - Declaration (in a .h file) of functions implemented in a .c file

    If your .cpp would call into one of those functions, Declaration need to be enclosed by

```
#ifdef __cplusplus
extern "C" {
#endif
        :   // functions declaration
#ifdef __cplusplus
}
#endif
```

Name mangling - Example

```
#ifdef __cplusplus
extern "C" {
#endif /* __cplusplus */

wchat * ANSIToUnicode(const char * str);
char * UnicodeToANSI(const wchar_t * str);
    :
#ifdef __cplusplus
}
#endif /* __cplusplus */
```

Case Study - Process Monitor Handler (1)

- Source File Problems
  - WISEAgent\Platform\Windows\platform.h
    - ✓ #define bool char
      #define true 1
      #define false 0
    - ✓ #if _MSC_VER < 1800 // <stdbool.h> already in VC++ 2013
      #define bool char
      #define true 1
      #define false 0
      #endif

Case Study - Process Monitor Handler (2)

- Source File Problems
  - ➢WISEAgent\Platform\Windows\common.c
    - ✓Unresolved ToolHelp APIs - CreateToolhelp32Snapshot, Process32First, Process32Next
      - Enumerate processes
    - ✓Call into CloseHandle to close snapshot
    - ✓Solution
      - Implements ToolHelp APIs in winsysstub.dll by kernel functionalities

        K32EnumProcesses

        K32GetModuleBaseName
      - Replace CloseHandle API call by CloseToolhelp32Snapshot if user want to close snapshot

Case Study - Process Monitor Handler (3)

- Toolhelp APIs
  - WISEAgent\Library3rdParty\winsysstub.winiotcore\impl\ToolHelp.cpp
- Token
  - WISEAgent\Library3rdParty\winsysstub.winiotcore\impl\Token.cpp
  - Adjust privilege before calling
    - ✓ OpenProcess
    - ✓ K32GetModuleBaseName
- Process
  - WISEAgent\Library3rdParty\winsysstub.winiotcore\impl\Processes.cpp
  - Convert vector of process id into vector of process handle

Case Study - Process Monitor Handler (4)

- Scope
  - WISEAgent\Library3rdParty\winsysstub.winiotcore\detail\Scope.h
  - Execute on exit scope
    - ✓ Use it in WISEAgent\Library3rdParty\winsysstub.winiotcore\impl\Processes.cpp
    - ✓ Adjusting privilege back on exit if we adjusted it before calling OpenProcess and K32GetModuleBaseName
- Unique Resource
  - WISEAgent\Library3rdParty\winsysstub.winiotcore\detail\UniqueResource.h
  - Not a perfected solution for unique resource
  - Refer to C++ N3949
    - ✓ http://www.open-std.org/jtc1/sc22/wg21/docs/papers/2014/n3949.pdf
  - Refer to Kenny Kerr modern C++ handle.h

Reference

Project	URL	Version	Source Tree Location in https://github.com/advlinda/cagentoniotcore
OpenSSL	https://www.openssl.org/	1.0.1h	WISEAgent\Library3rdParty\openssl-1.0.1h
Curl	http://curl.haxx.se/	7.37.1	WISEAgent\Library3rdParty\curl-7.37.1
Libxml2	http://www.xmlsoft.org/	2.7.8	WISEAgent\Library3rdParty\libxml2-2.7.8
Pthreads Win32	https://www.sourceware.org/pthreads-win32/		WISEAgent\Library3rdParty\pthread.win32
mosquitto	http://mosquitto.org	1.3.4	WISEAgent\Library3rdParty\mosquito-1.3.4

Calls to Action

Join WinHEC LINE Community
@winhec

We want to hear from you!

Please Complete the Evaluation Form and return it to our reception.

Your input is highly important to us!
Thank you!! ☺

Microsoft

© 2016 Microsoft Corporation. All rights reserved.