RCA - CVE-2017-12561

Analysis through binary diffing using Ghidra. The idea is trying to locate where the bug was located by diffing the patched version with an earlier version of the software.

iMC PLAT 7.3E0504 and iMC PLAT 7.3E0504P04

Introduction

The structure of this document relies to some extent on: https://raw.githubusercontent.com/googleprojectzero/0days-in-the-wild/main/0day-RCAs/template.md

The Basics

Disclosure

- 2017-06-27 - Vulnerability reported to vendor
- 2017-10-03 - Coordinated public release of advisory

Product:

Hewlett Packard Enterprise Intelligent Management Center

Affected Versions:

HPE IMC v7.3 (E0504)

Reporter:

Steven Seeley (mr_me) of Offensive Security

The Vulnerability

Bug class:

RCE(Remote Code Execution) through UAF(Use-After-Free)

dbman Opcode 10012 Use-After-Free Remote Code Execution

Vulnerability details:

https://securitytracker.com/id/1039495
A remote authenticated user can send specially crafted data to the dbman service on TCP port 2810 to trigger a use-after-free in the mibFileServlet servlet to delete arbitrary files on the target system [CVE-2017-12561].
This resource seems to require a user to be authenticated to trigger the vulnerability.

https://www.zerodayinitiative.com/advisories/ZDI-17-836/
This resource states that authentication is not required to exploit this vulnerability.

From my tinkering so far, it seems that the authentication is not required. But default firewall rules on my test environment(WinServer2012R2) prevent reaching dbman port 2810 i.e., prevent sending payloads to it.

Analysis - Process

Started osinting to find out what is out there about this vulnerability CVE-2017-12561. There are a number of vulnerabilities related to this one.

OSINT

[ZDI-17-836 / CVE-12561]

- A bunch of links for vulnerabilities revolving around this one,
  - Querying some of this sites gives an idea:
    - exploitdb
    - cvedetails
    - cvesearch
    - https://vulners.com/nessus/HP_INTELLIGENT_MANAGEMENT_CENTER_7_3_E0504P04.NASL
    - https://cxsecurity.com/cveproduct/7/18987/intelligent_management_center/

Structure of the analysis - technical

Here is the high level view of how I am approaching this stuff(excluding preliminary reading/analysis/osinting):

0. Setup the hosting environment

1. Setup the software (Hewlett Packard Enterprise Intelligent Management Center)
2. Try to confirm the bug
3. Refine previous step

Environment Setup - Installation

The following are the steps that I've taken to deploy the iMC software.

Host

My host is:

```
$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 20.04.3 LTS
Release:    20.04
Codename:   focal
```

I am using a type 2 hypervisor: Oracle VirtualBox, version/package:

```
$ dpkg-query -W virtualbox
virtualbox  6.1.26-dfsg-3~ubuntu1.20.04.
```

This is the version from Canonical's repo, not directly from Oracle.

Test Environment

I am using:

- **OS**: MS Windows Sever 2012 R2
- **Additional software for DB**: DotNet 4.5.2
- **Software DB**: SQLServer2014Express
- **Software**: iMC_PLAT_7.3_E0504_Ent_Win-003(vuln)
- **Software**: iMC_PLAT_7.3_E0504P04_Win-002(patched)

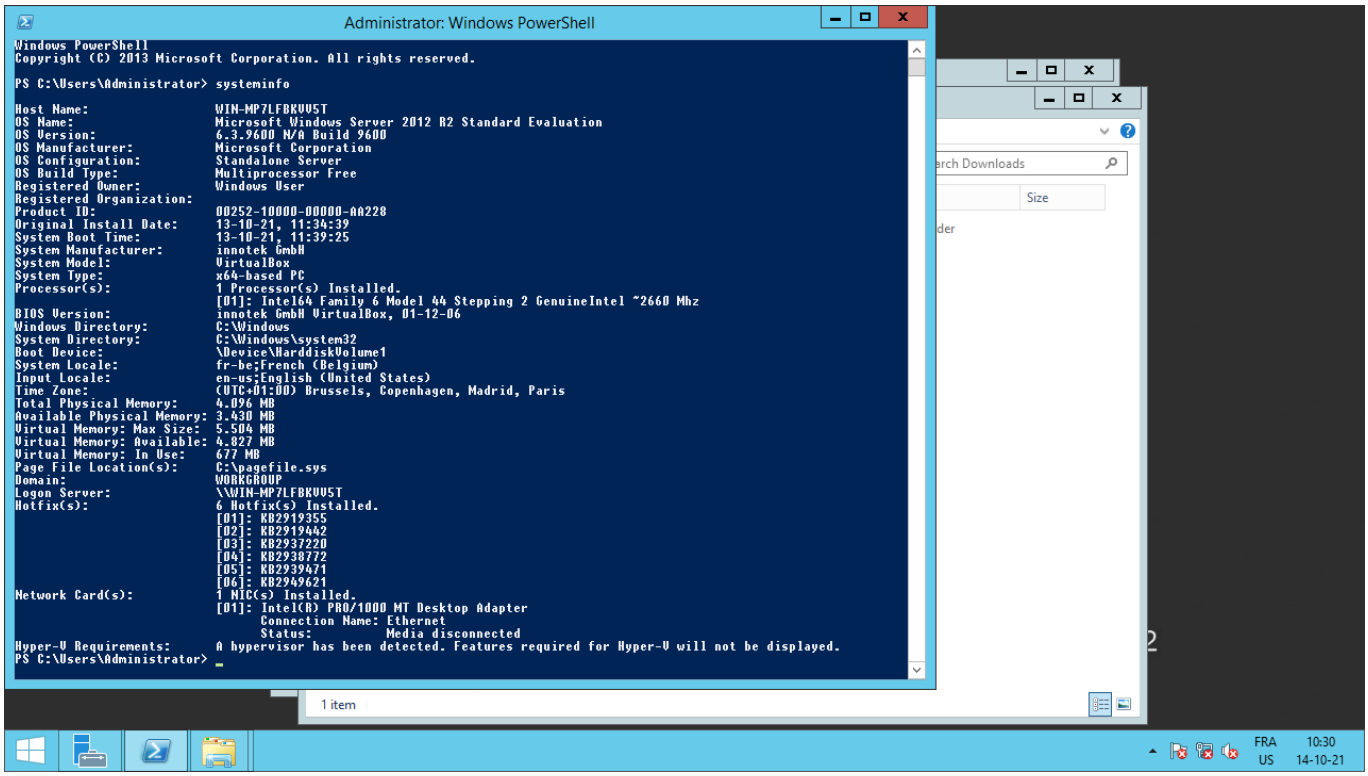
After deeper analysis, the version E0504 is affected by a stack-based buffer overflow or stack buffer overrun in msecdbg nomenclature. Whereas version E0504P04 introduces a potential remote code execution(RCE) in a specific function.

OS:

MS Windows Sever 2012 R2:

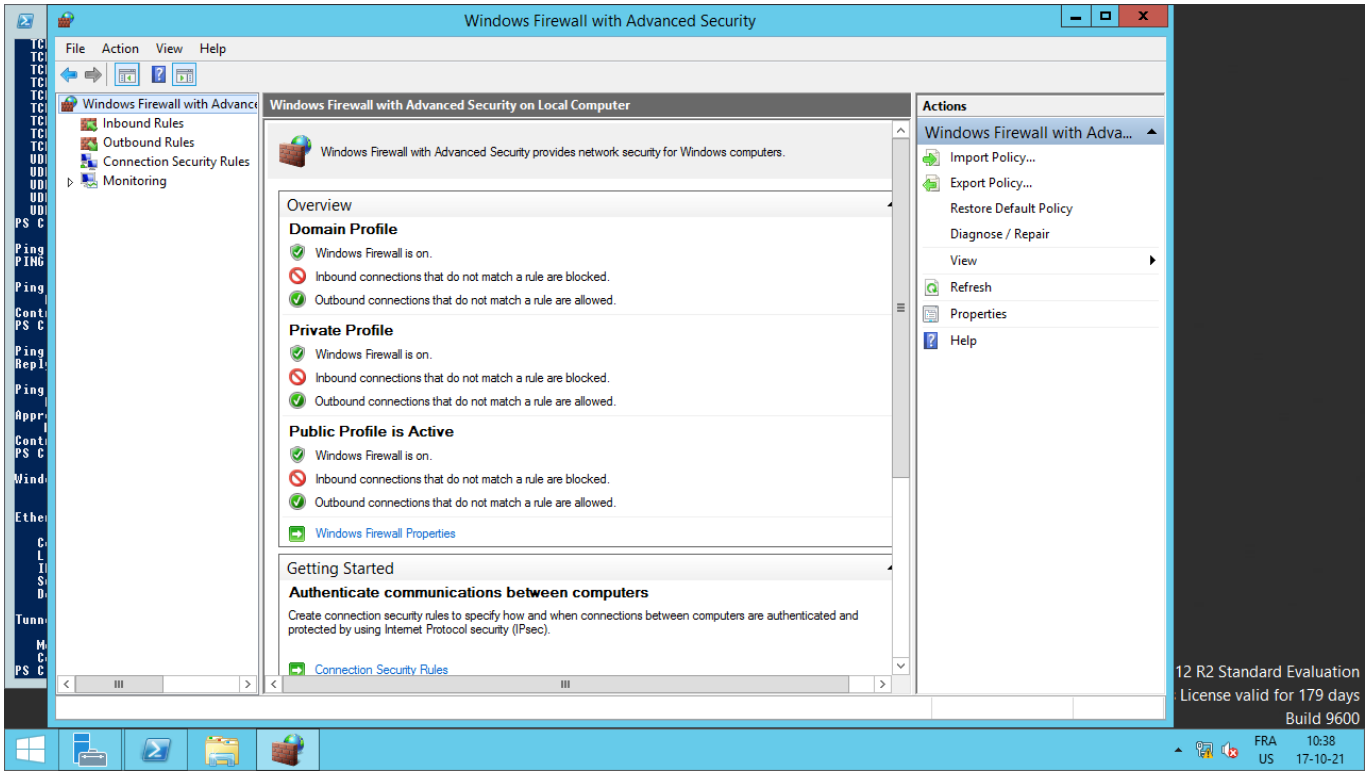
```
$ sha1sum 9600.17050.WINBLUE_REFRESH-140317-1640_X64FRE_SERVER_EVAL_EN-US-IR3_SSS_X64FREE_EN-US_DV9.ISO
849734f37346385dac2c101e4aacba4626bb141c  9600.17050.WINBLUE_REFRESH-140317-1640_X64FRE_SERVER_EVAL_EN-US-IR3_SSS_X64FREE_EN-US_DV9.ISO
```

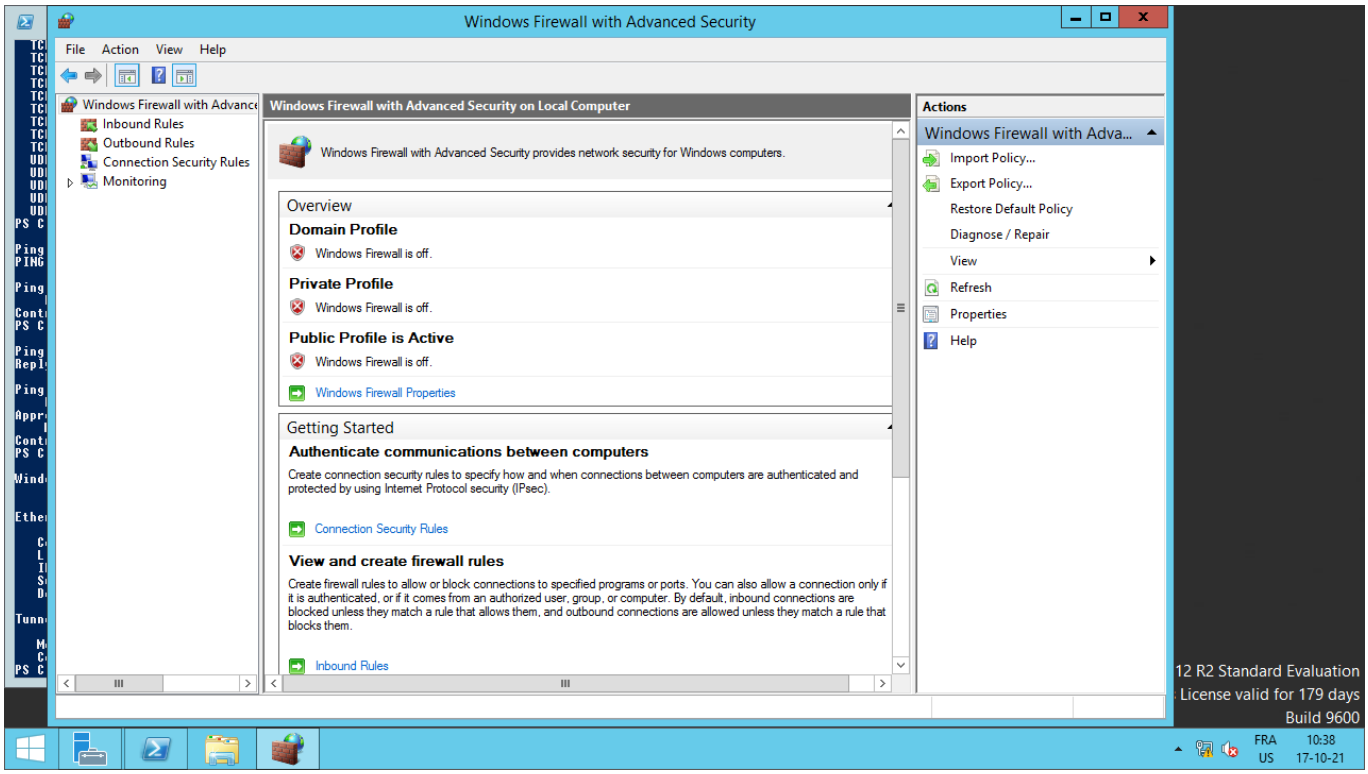
The installation process is all standard, nothing fancy. I just avoided enabling Internet connectivity for old/"legacy" systems. One exception to this was during the installation of SQLServer2014.



[systeminfo]

By default the firewall is enabled on the three profiles: domain, private, public. If we want to reach *dbman* we need to disable the firewall or add a specific rule for it.





All red, we're good to go.

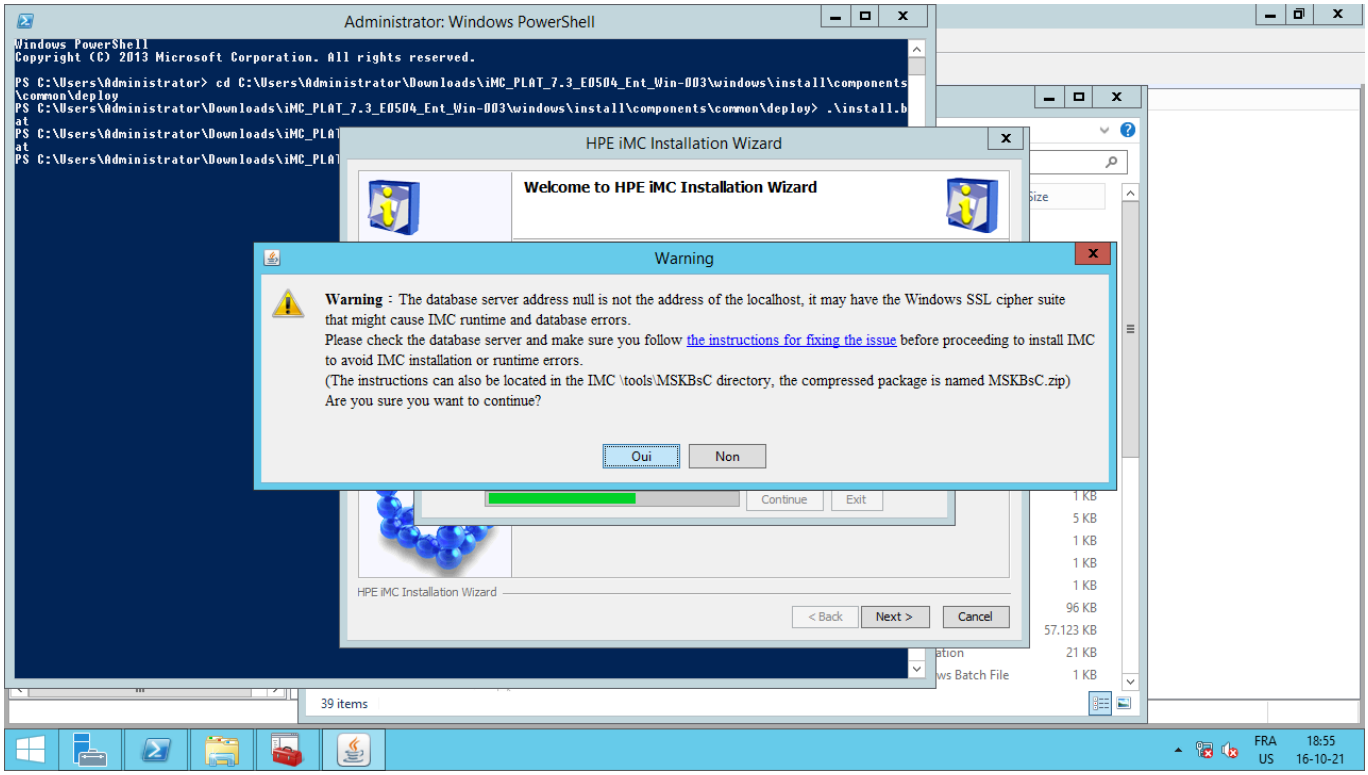
Software:

DotNet 4.5.2:

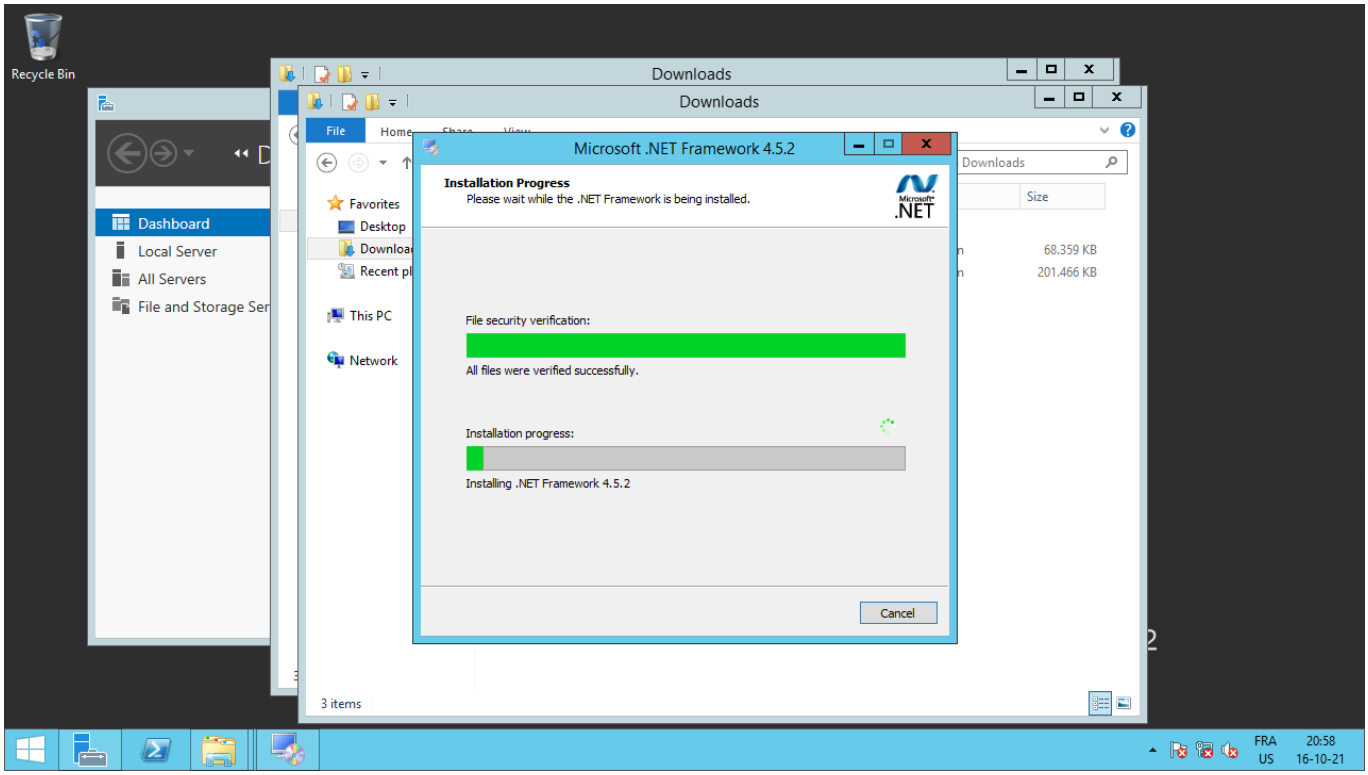
```
$ sha1sum NDP452-KB2901907-x86-x64-AllOS-ENU.exe
89f86f9522dc7a8a965facce839abb790a285a63  NDP452-KB2901907-x86-x64-AllOS-ENU.exe
```

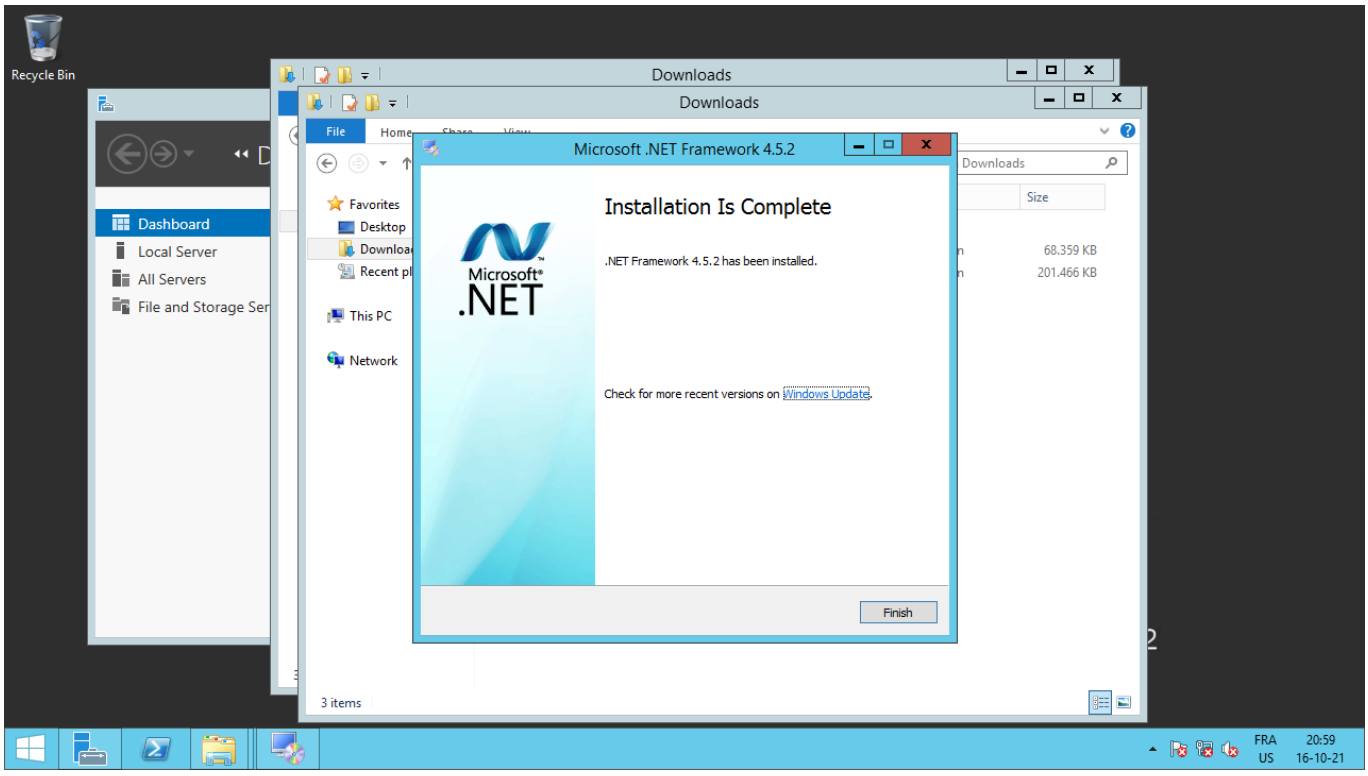
https://dotnet.microsoft.com/download/dotnet-framework/net452

This is needed in order to avoid SSL/TLS error when launching the iMC installer. In my experiments the iMC installer database check fails if there isn't a bunch of updates ; something like:



Installation steps:



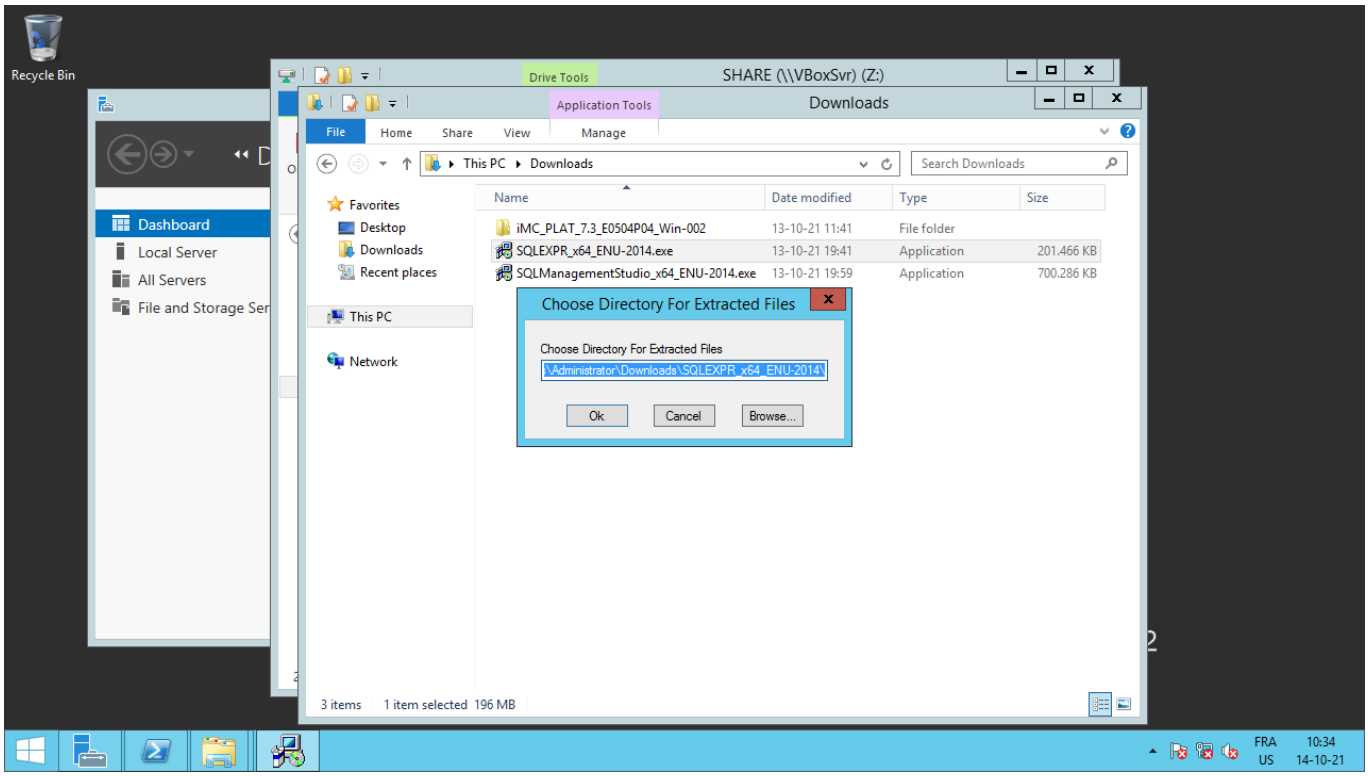


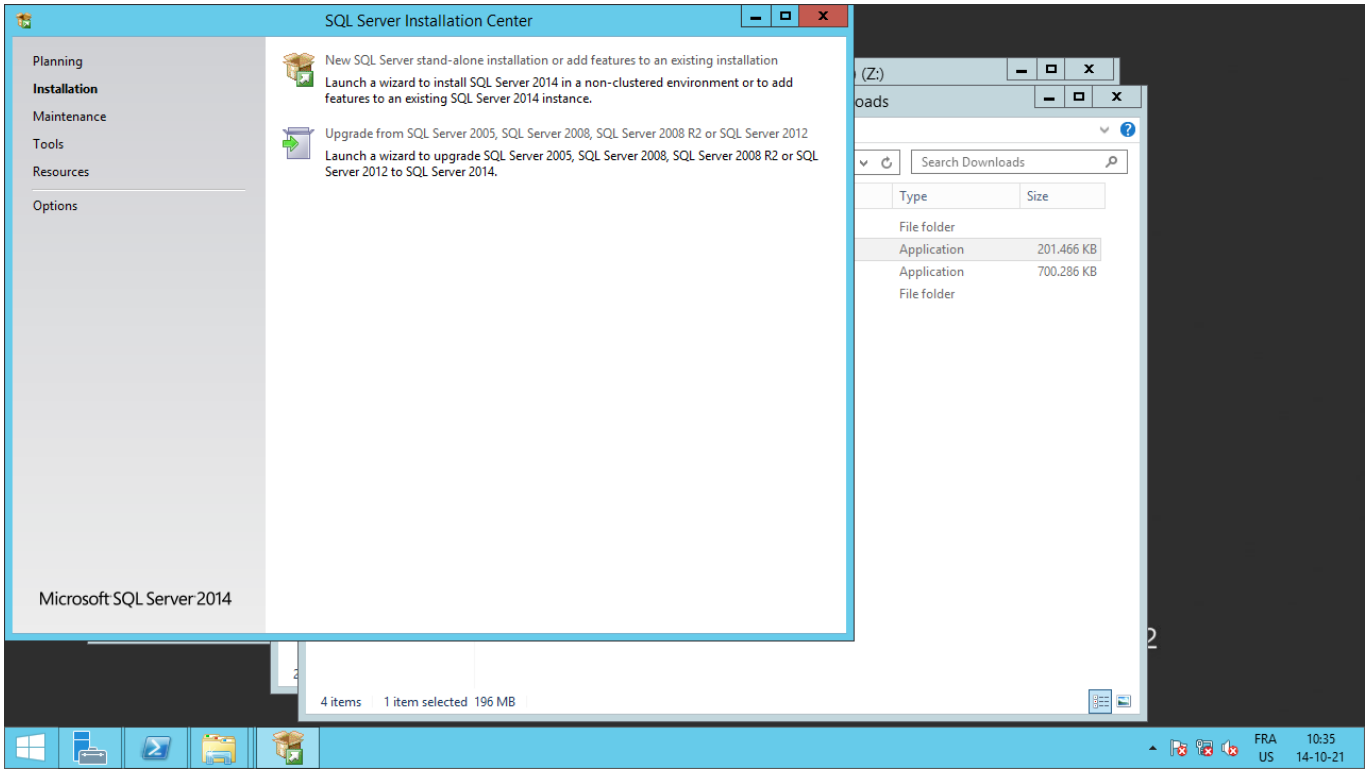
SQLServer2014Express:

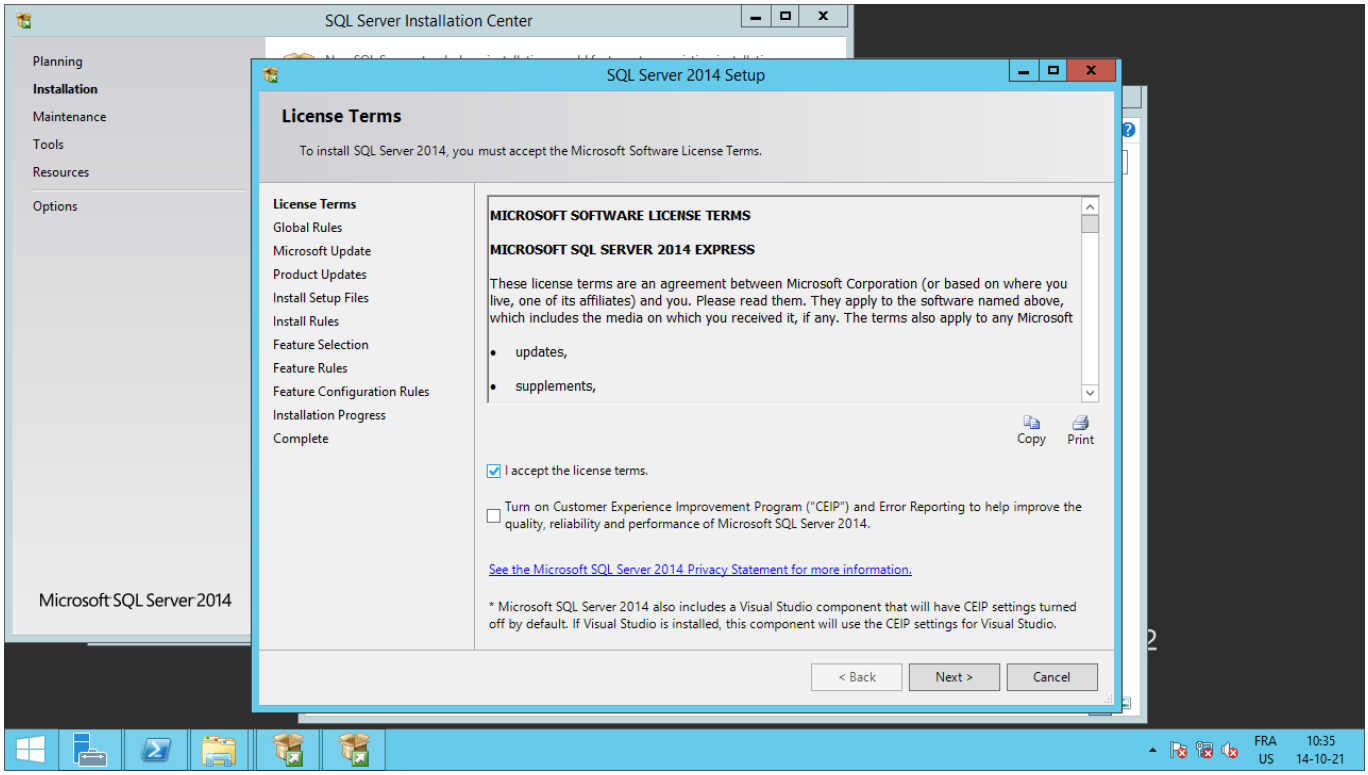
```
$ sha1sum SQLEXPR_x64_ENU-2014.exe
38e0d08f06af2f907e1aebdca0444b828ae356b5  SQLEXPR_x64_ENU-2014.exe
```

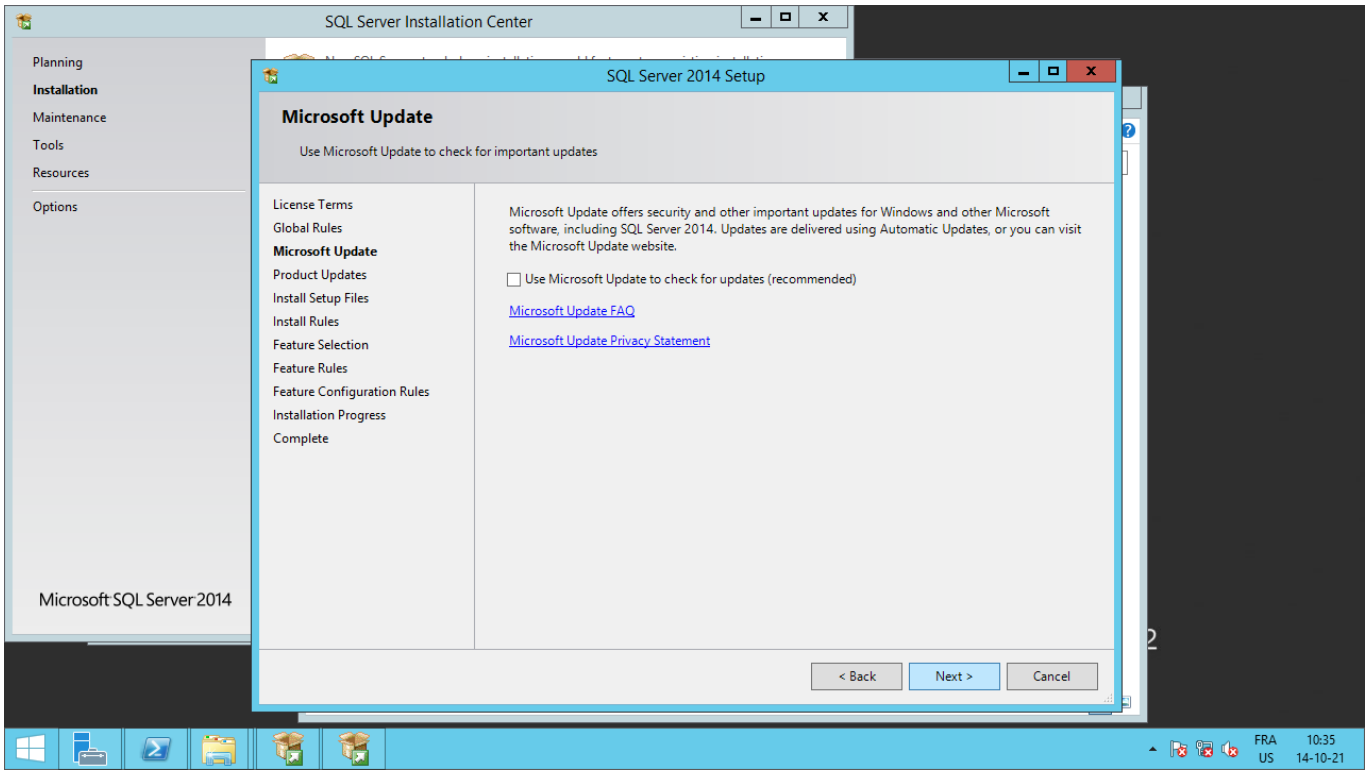
https://www.microsoft.com/en-us/download/details.aspx?id=42299

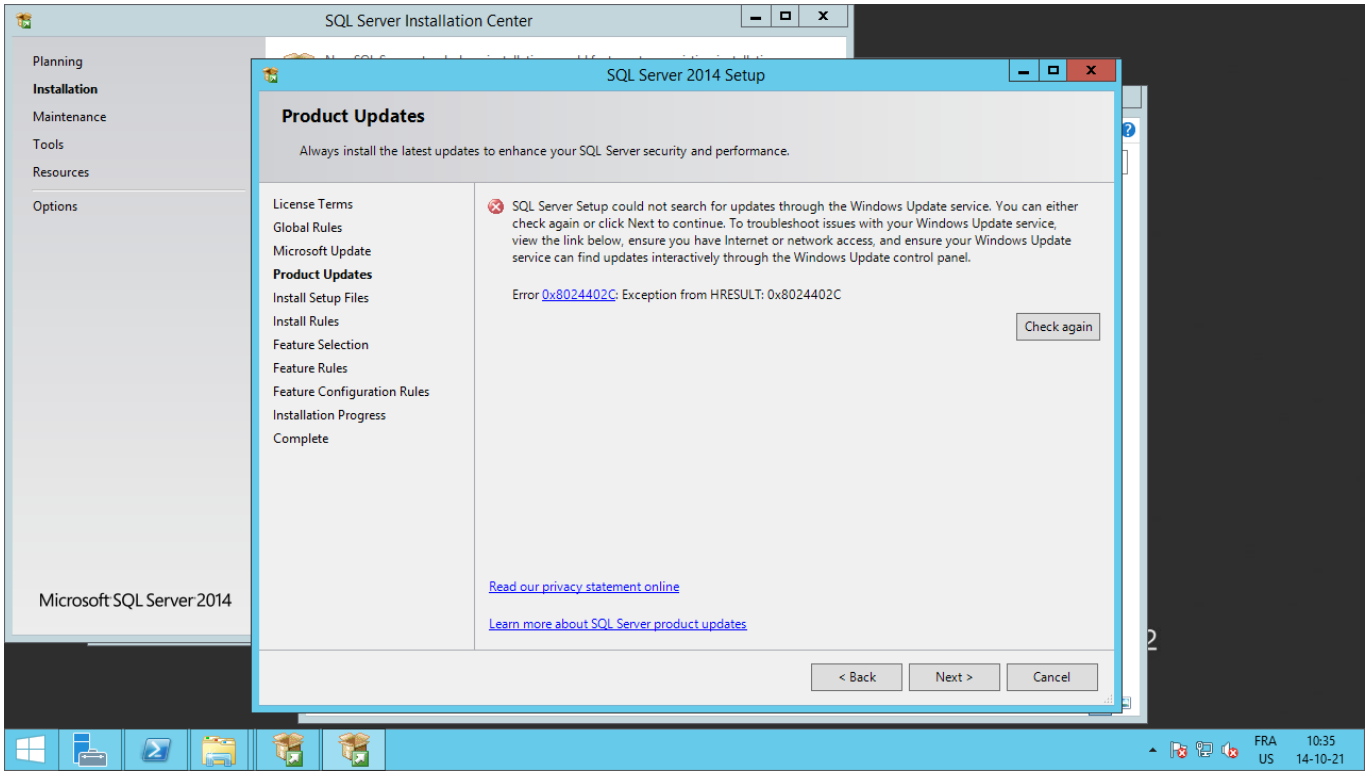
Installation steps:



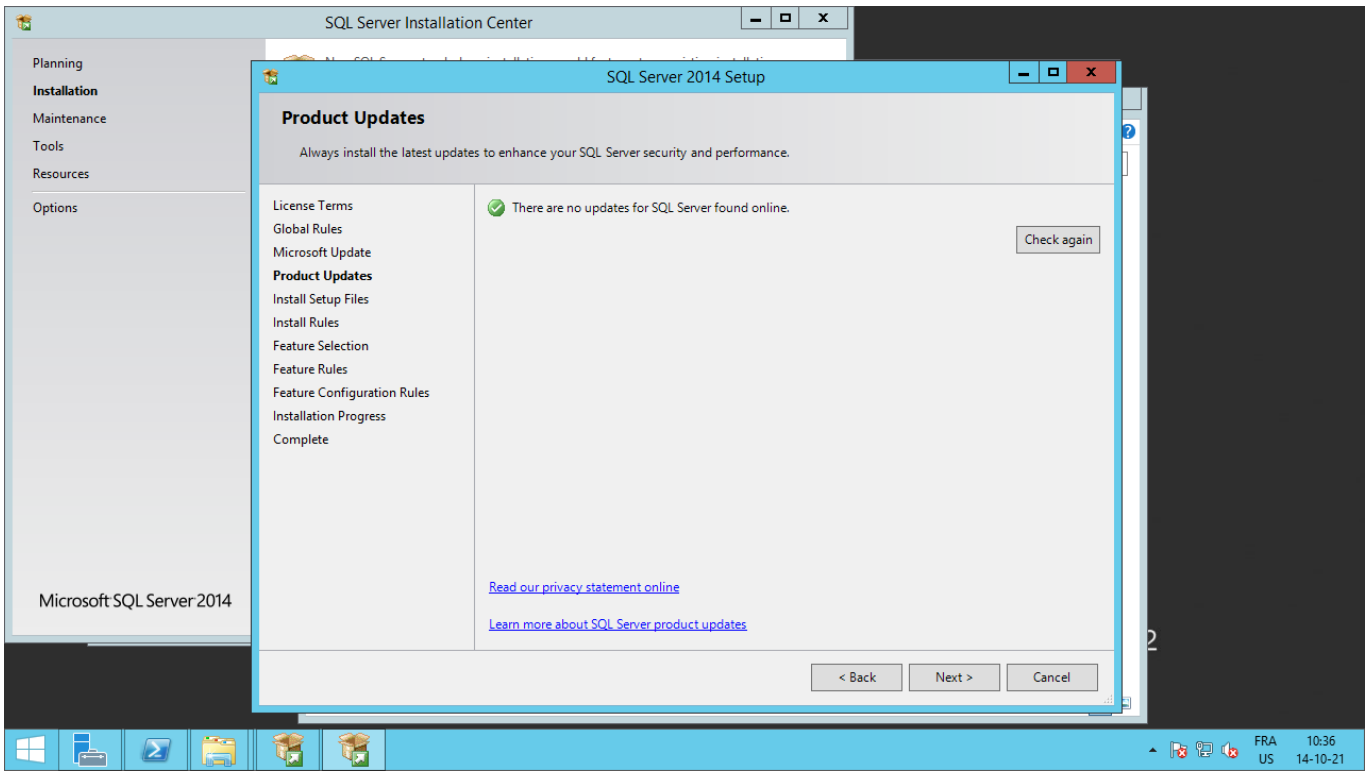


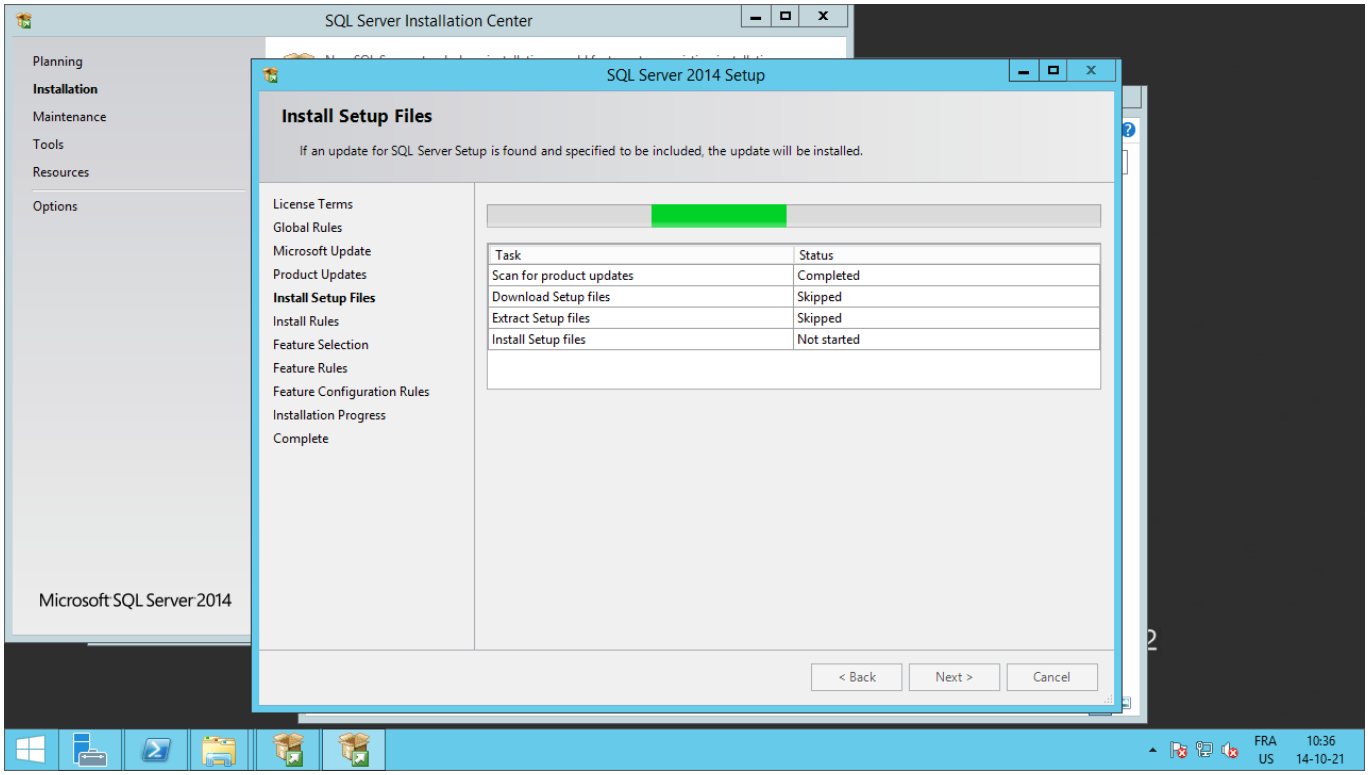

SQL Server Installation Center
Planning
Installation
Maintenance
Tools
Resources
Options
Microsoft SQL Server 2014
SQL Server 2014 Setup
License Terms
To install SQL Server 2014, you must accept the Microsoft Software License Terms.
License Terms
Global Rules
Microsoft Update
Product Updates
Install Setup Files
Install Rules
Feature Selection
Feature Rules
Feature Configuration Rules
Installation Progress
Complete
MICROSOFT SOFTWARE LICENSE TERMS
MICROSOFT SQL SERVER 2014 EXPRESS
These license terms are an agreement between Microsoft Corporation (or based on where you live, one of its affiliates) and you. Please read them. They apply to the software named above, which includes the media on which you received it, if any. The terms also apply to any Microsoft
updates,
supplements,
Copy
Print
I accept the license terms.
Turn on Customer Experience Improvement Program ("CEIP") and Error Reporting to help improve the quality, reliability and performance of Microsoft SQL Server 2014.
See the Microsoft SQL Server 2014 Privacy Statement for more information.
* Microsoft SQL Server 2014 also includes a Visual Studio component that will have CEIP settings turned off by default. If Visual Studio is installed, this component will use the CEIP settings for Visual Studio.
< Back
Next >
Cancel
FRA
US
10:35
14-10-21

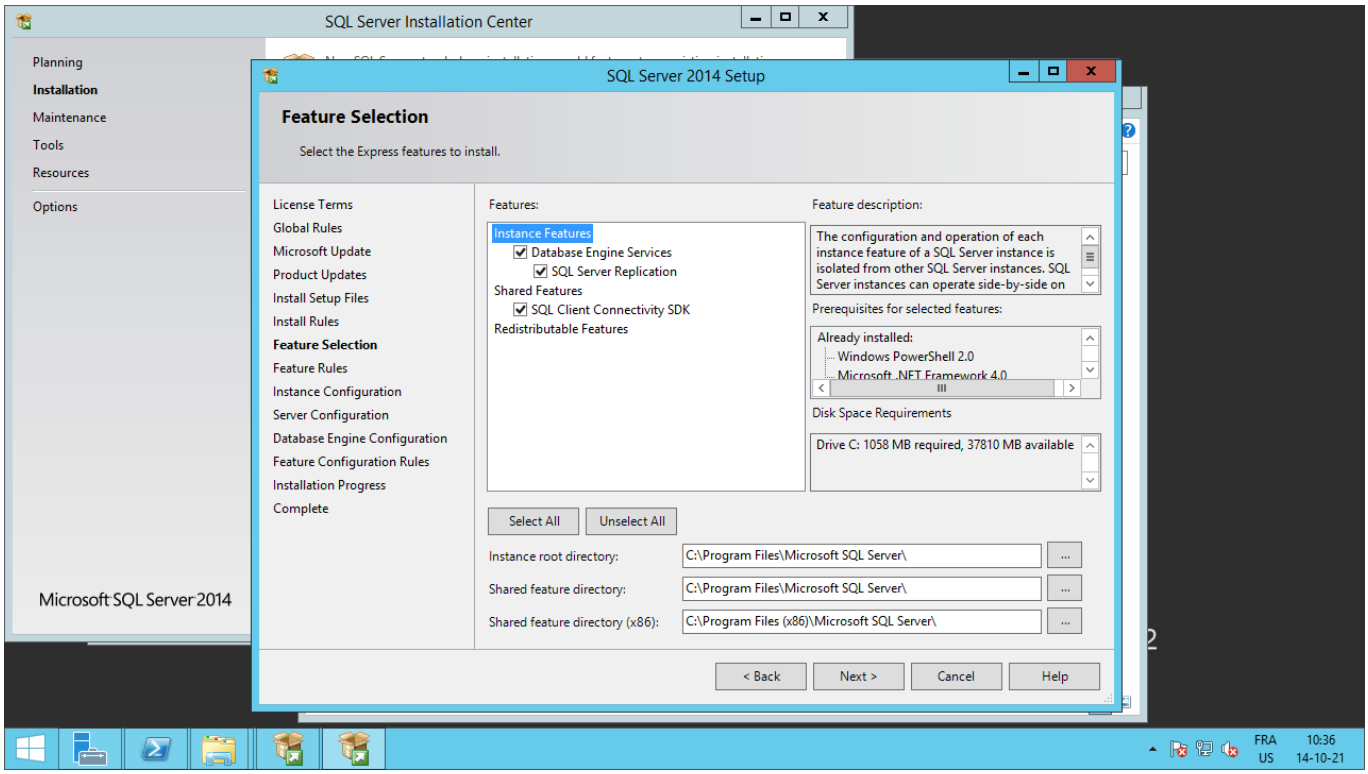

SQL Server Installation Center
Planning
Installation
Maintenance
Tools
Resources
Options
Microsoft SQL Server 2014
SQL Server 2014 Setup
Microsoft Update
Use Microsoft Update to check for important updates
License Terms
Global Rules
Microsoft Update
Product Updates
Install Setup Files
Install Rules
Feature Selection
Feature Rules
Feature Configuration Rules
Installation Progress
Complete
Microsoft Update offers security and other important updates for Windows and other Microsoft software, including SQL Server 2014. Updates are delivered using Automatic Updates, or you can visit the Microsoft Update website.
Use Microsoft Update to check for updates (recommended)
Microsoft Update FAQ
Microsoft Update Privacy Statement
< Back
Next >
Cancel
FRA
US
10:35
14-10-21

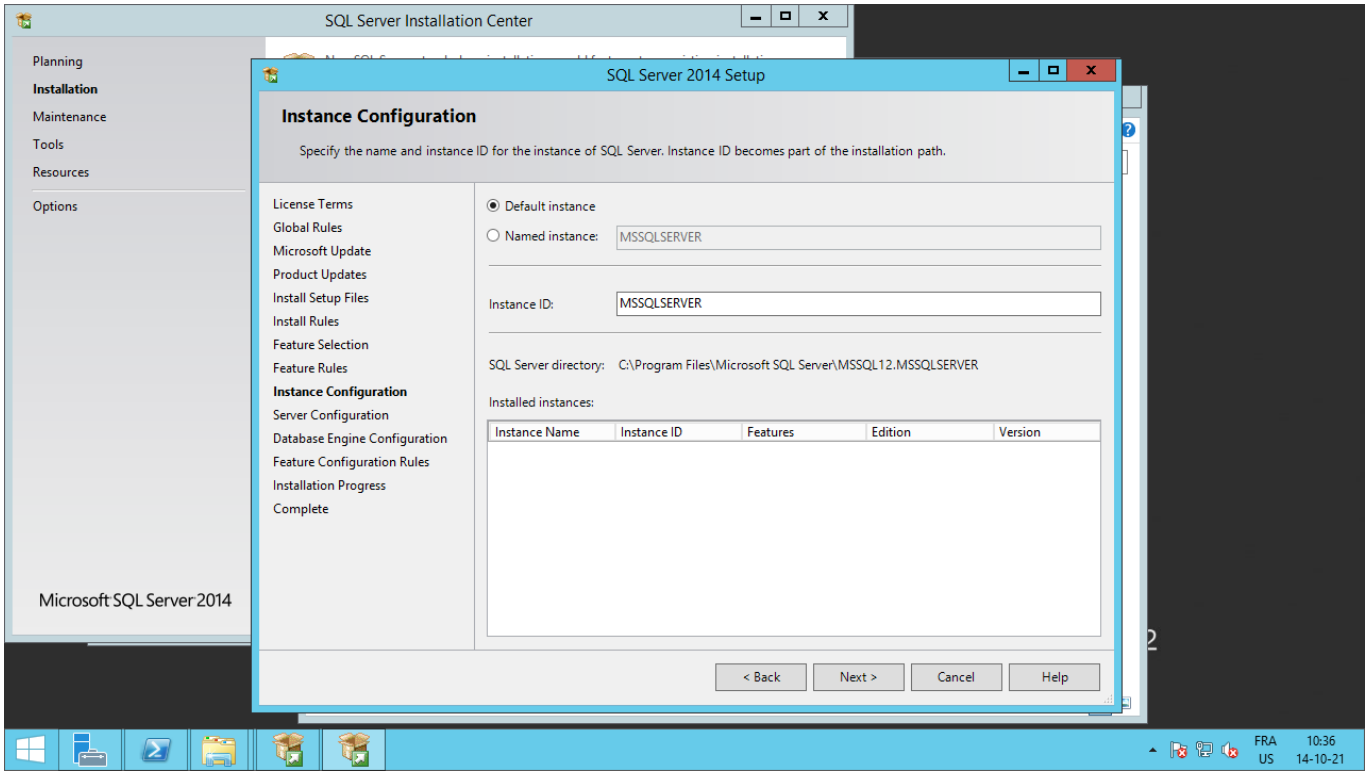


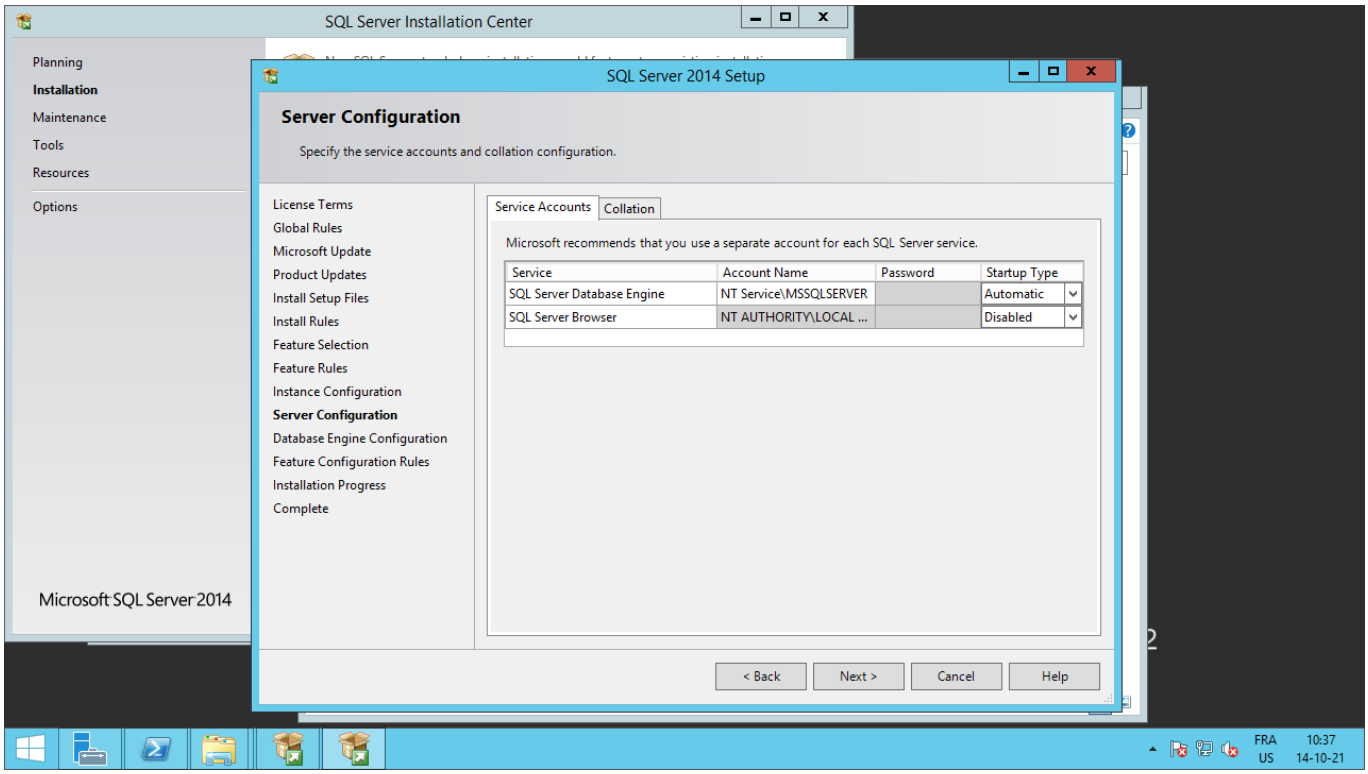
At this point, I enable Internet connectivity so that it can perform the check and also needed later on for the "NetFx3" install.

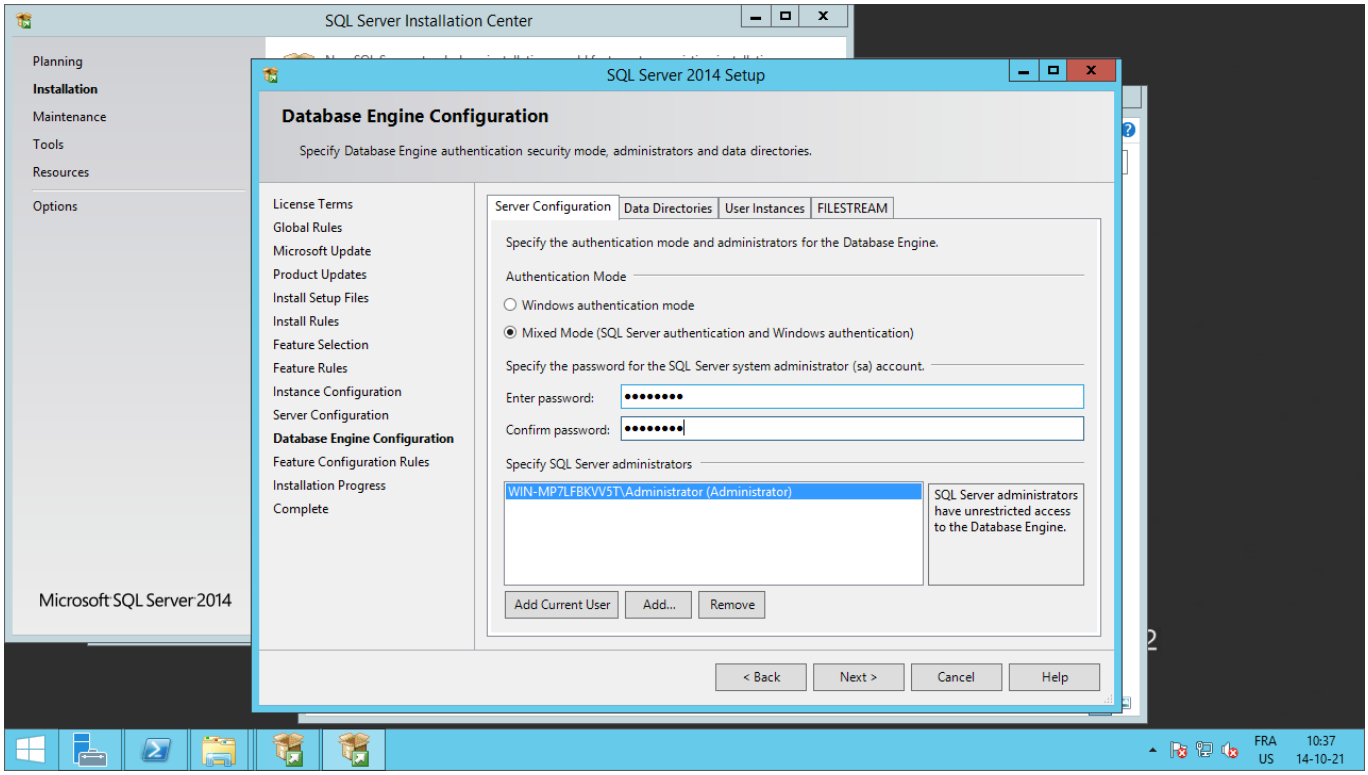


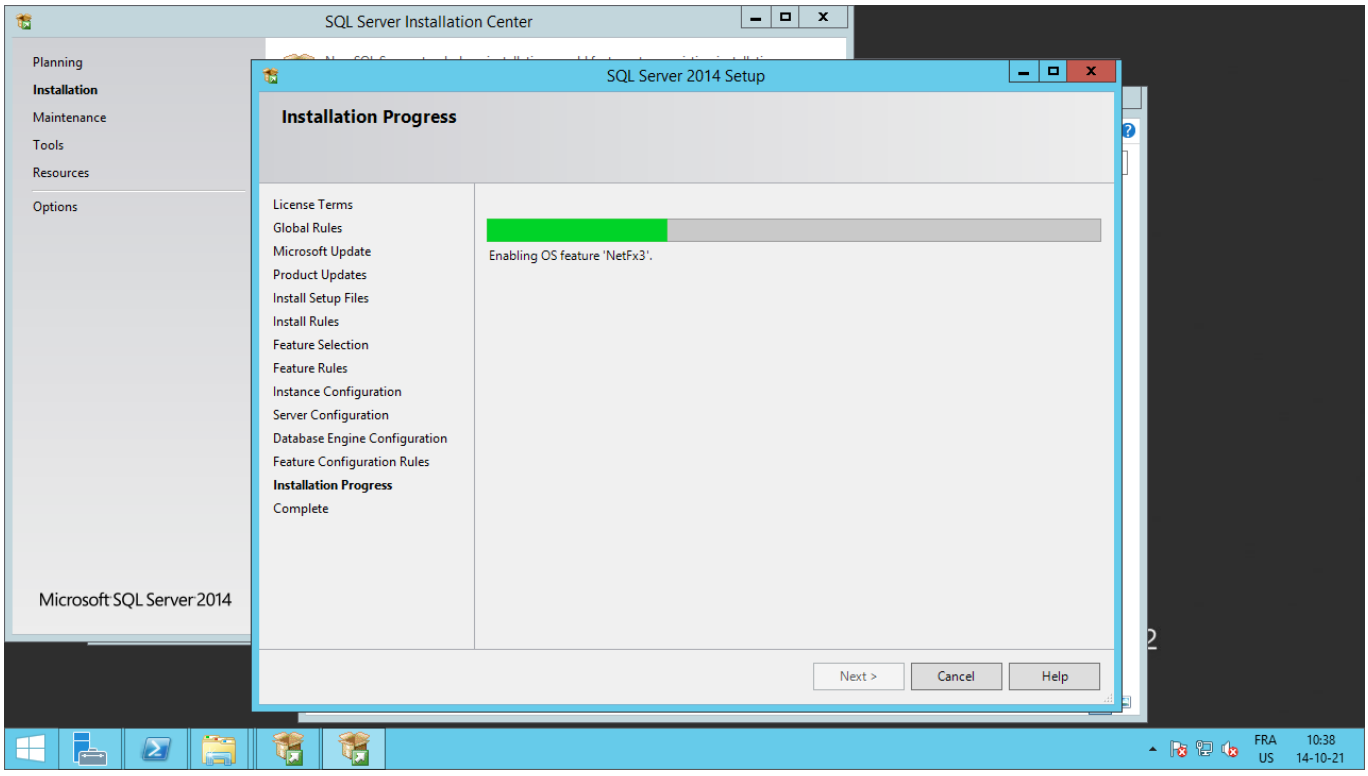

SQL Server Installation Center
Planning
Installation
Maintenance
Tools
Resources
Options
Microsoft SQL Server 2014
SQL Server 2014 Setup
Install Setup Files
If an update for SQL Server Setup is found and specified to be included, the update will be installed.
License Terms
Global Rules
Microsoft Update
Product Updates
Install Setup Files
Install Rules
Feature Selection
Feature Rules
Feature Configuration Rules
Installation Progress
Complete
Task
Status
Scan for product updates
Completed
Download Setup files
Skipped
Extract Setup files
Skipped
Install Setup files
Not started
< Back
Next >
Cancel
FRA
US
10:36
14-10-21

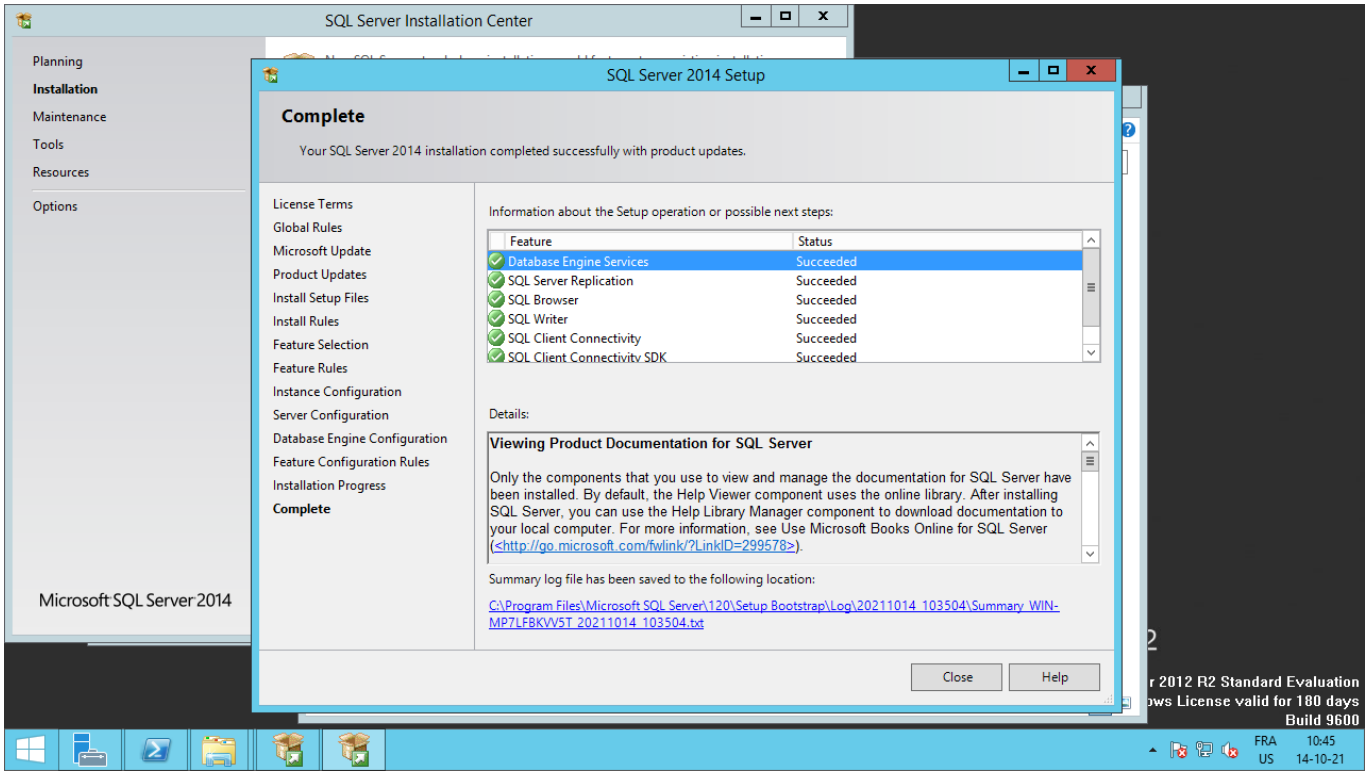

SQL Server Installation Center
Planning
Installation
Maintenance
Tools
Resources
Options
Microsoft SQL Server 2014
SQL Server 2014 Setup
Feature Selection
Select the Express features to install.
License Terms
Global Rules
Microsoft Update
Product Updates
Install Setup Files
Install Rules
Feature Selection
Feature Rules
Instance Configuration
Server Configuration
Database Engine Configuration
Feature Configuration Rules
Installation Progress
Complete
Features:
Instance Features
Database Engine Services
SQL Server Replication
Shared Features
SQL Client Connectivity SDK
Redistributable Features
Feature description:
The configuration and operation of each instance feature of a SQL Server instance is isolated from other SQL Server instances. SQL Server instances can operate side-by-side on
Prerequisites for selected features:
Already installed:
Windows PowerShell 2.0
Microsoft .NET Framework 4.0
Disk Space Requirements
Drive C: 1058 MB required, 37810 MB available
Select All
Unselect All
Instance root directory:
C:\Program Files\Microsoft SQL Server\
Shared feature directory:
C:\Program Files\Microsoft SQL Server\
Shared feature directory (x86):
C:\Program Files (x86)\Microsoft SQL Server\
< Back
Next >
Cancel
Help
FRA
US
10:36
14-10-21


SQL Server Installation Center
Planning
Installation
Maintenance
Tools
Resources
Options
Microsoft SQL Server 2014
SQL Server 2014 Setup
Instance Configuration
Specify the name and instance ID for the instance of SQL Server. Instance ID becomes part of the installation path.
License Terms
Global Rules
Microsoft Update
Product Updates
Install Setup Files
Install Rules
Feature Selection
Feature Rules
Instance Configuration
Server Configuration
Database Engine Configuration
Feature Configuration Rules
Installation Progress
Complete
Default instance
Named instance:
MSSQLSERVER
Instance ID:
MSSQLSERVER
SQL Server directory:
C:\Program Files\Microsoft SQL Server\MSSQL12.MSSQLSERVER
Installed instances:
Instance Name
Instance ID
Features
Edition
Version
< Back
Next >
Cancel
Help
FRA
US
10:36
14-10-21

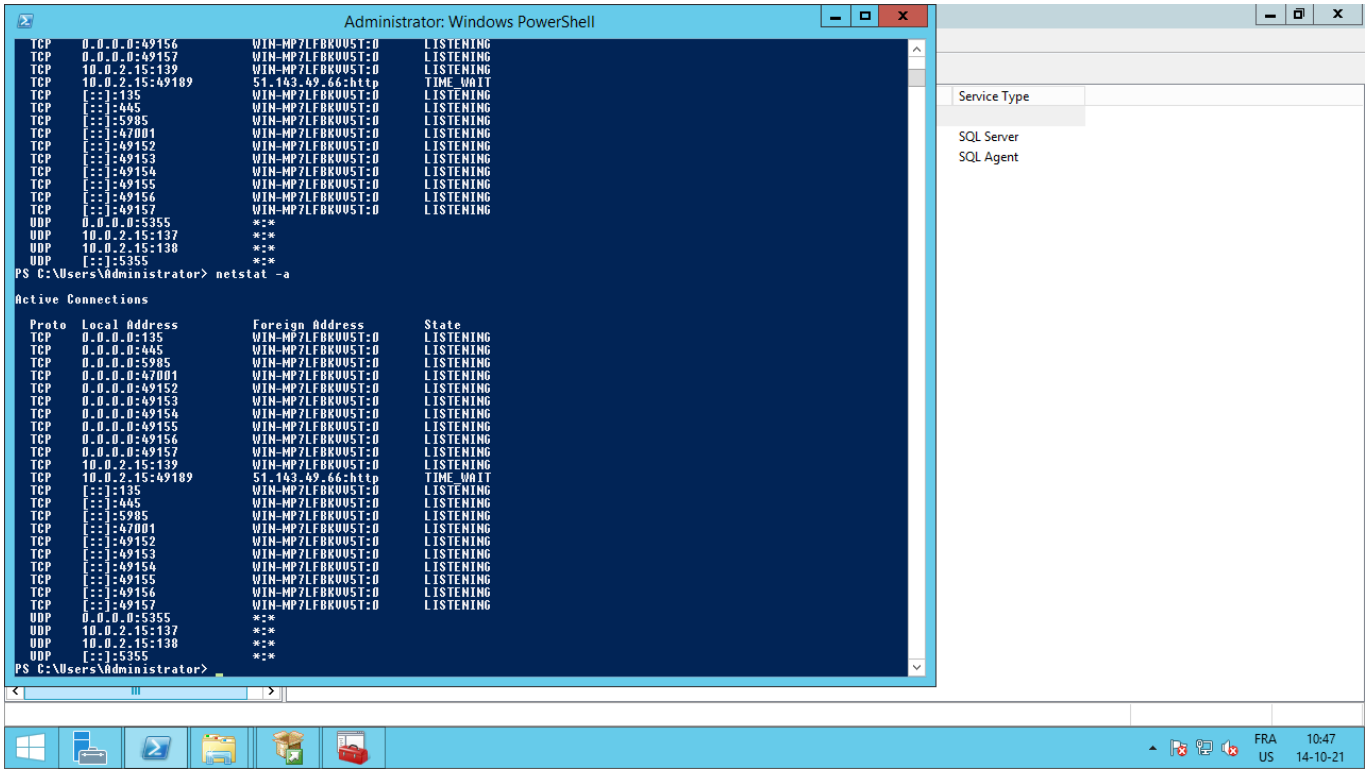

SQL Server Installation Center
Planning
Installation
Maintenance
Tools
Resources
Options
Microsoft SQL Server 2014
SQL Server 2014 Setup
Server Configuration
Specify the service accounts and collation configuration.
License Terms
Global Rules
Microsoft Update
Product Updates
Install Setup Files
Install Rules
Feature Selection
Feature Rules
Instance Configuration
Server Configuration
Database Engine Configuration
Feature Configuration Rules
Installation Progress
Complete
Service Accounts
Collation
Microsoft recommends that you use a separate account for each SQL Server service.
Service
Account Name
Password
Startup Type
SQL Server Database Engine
NT Service\MSSQLSERVER
Automatic
SQL Server Browser
NT AUTHORITY\LOCAL ...
Disabled
< Back
Next >
Cancel
Help
FRA
US
10:37
14-10-21


SQL Server Installation Center
Planning
Installation
Maintenance
Tools
Resources
Options
Microsoft SQL Server 2014
SQL Server 2014 Setup
Database Engine Configuration
Specify Database Engine authentication security mode, administrators and data directories.
License Terms
Global Rules
Microsoft Update
Product Updates
Install Setup Files
Install Rules
Feature Selection
Feature Rules
Instance Configuration
Server Configuration
Database Engine Configuration
Feature Configuration Rules
Installation Progress
Complete
Server Configuration
Data Directories
User Instances
FILESTREAM
Specify the authentication mode and administrators for the Database Engine.
Authentication Mode
Windows authentication mode
Mixed Mode (SQL Server authentication and Windows authentication)
Specify the password for the SQL Server system administrator (sa) account.
Enter password:
Confirm password:
Specify SQL Server administrators
WIN-MP7LFBKVV5T\Administrator (Administrator)
SQL Server administrators have unrestricted access to the Database Engine.
Add Current User
Add...
Remove
< Back
Next >
Cancel
Help
FRA
US
10:37
14-10-21

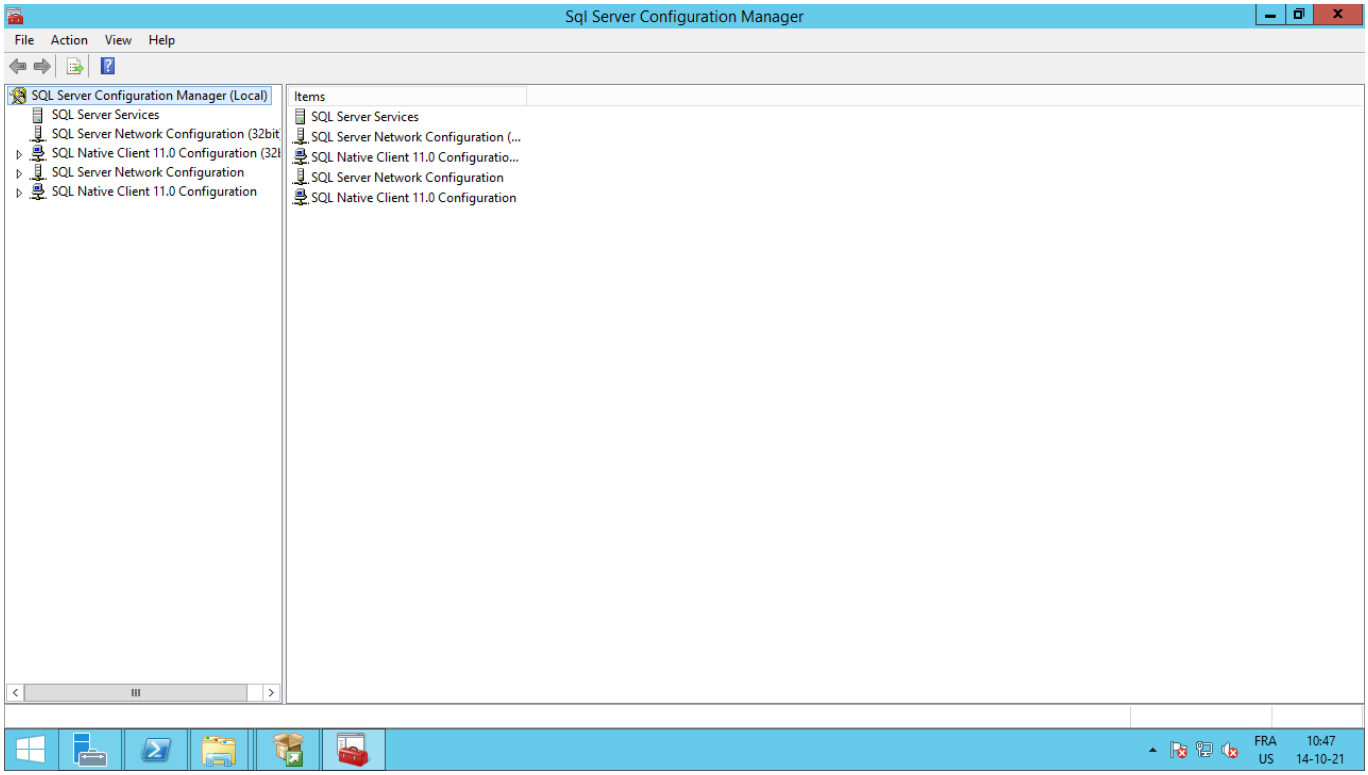

SQL Server Installation Center
Planning
Installation
Maintenance
Tools
Resources
Options
Microsoft SQL Server 2014
SQL Server 2014 Setup
Installation Progress
License Terms
Global Rules
Microsoft Update
Product Updates
Install Setup Files
Install Rules
Feature Selection
Feature Rules
Instance Configuration
Server Configuration
Database Engine Configuration
Feature Configuration Rules
Installation Progress
Complete
Enabling OS feature 'NetFx3'.
Next >
Cancel
Help
FRA
US
10:38
14-10-21

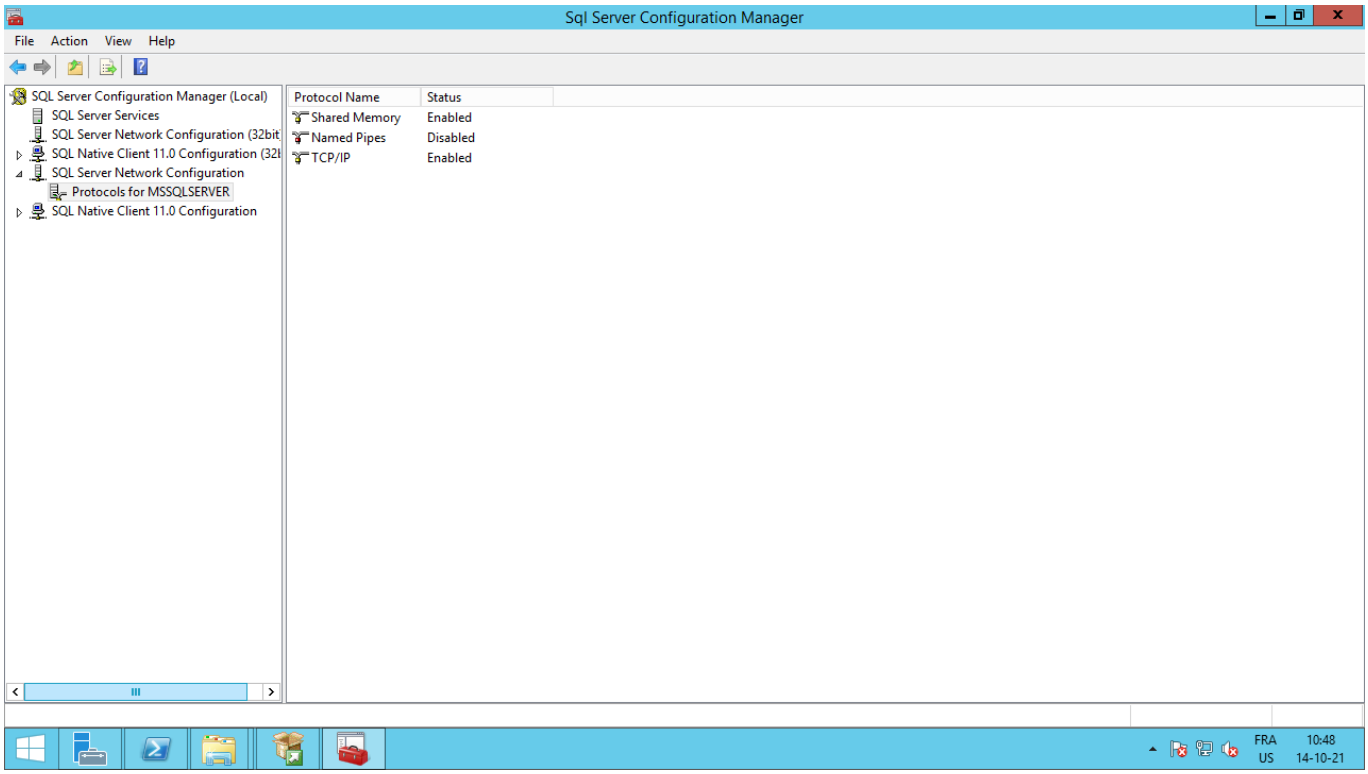


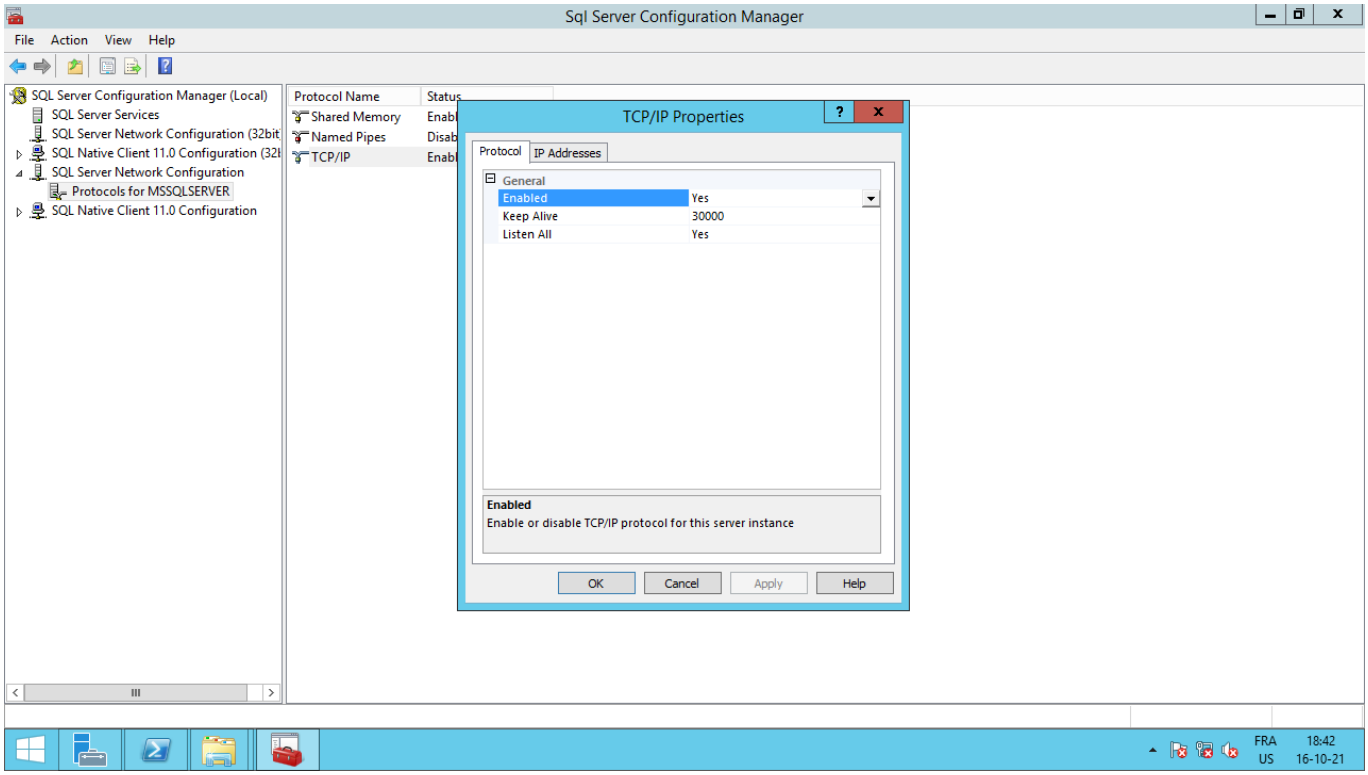
At this point, the db is installed but not yet exposed on port 1433:

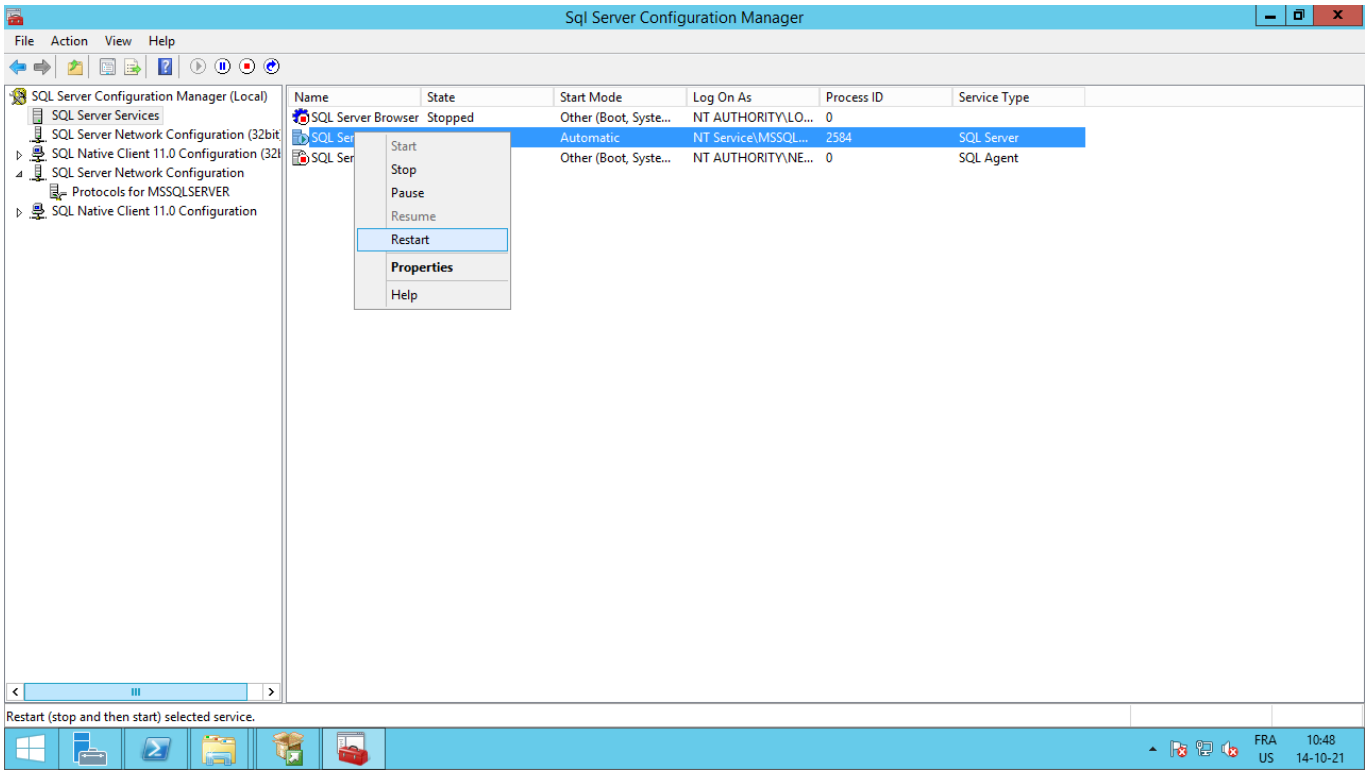


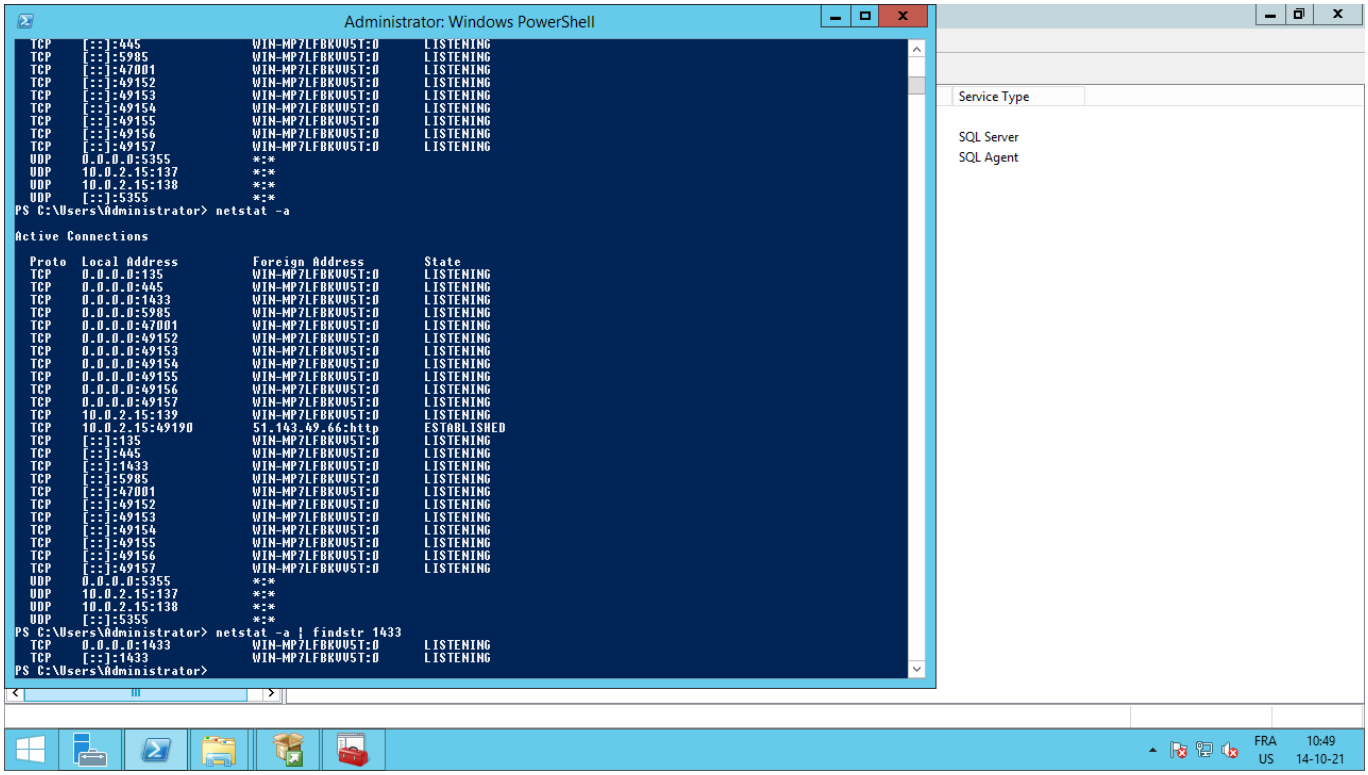
This can be achieved by enabling TCP/IP through the *SQL Server Configuration Manager*:


Sql Server Configuration Manager
File Action View Help
SQL Server Configuration Manager (Local)
SQL Server Services
SQL Server Network Configuration (32bit
SQL Native Client 11.0 Configuration (32
SQL Server Network Configuration
SQL Native Client 11.0 Configuration
Items
SQL Server Services
SQL Server Network Configuration (...
SQL Native Client 11.0 Configuratio...
SQL Server Network Configuration
SQL Native Client 11.0 Configuration
FRA
US
10:47
14-10-21

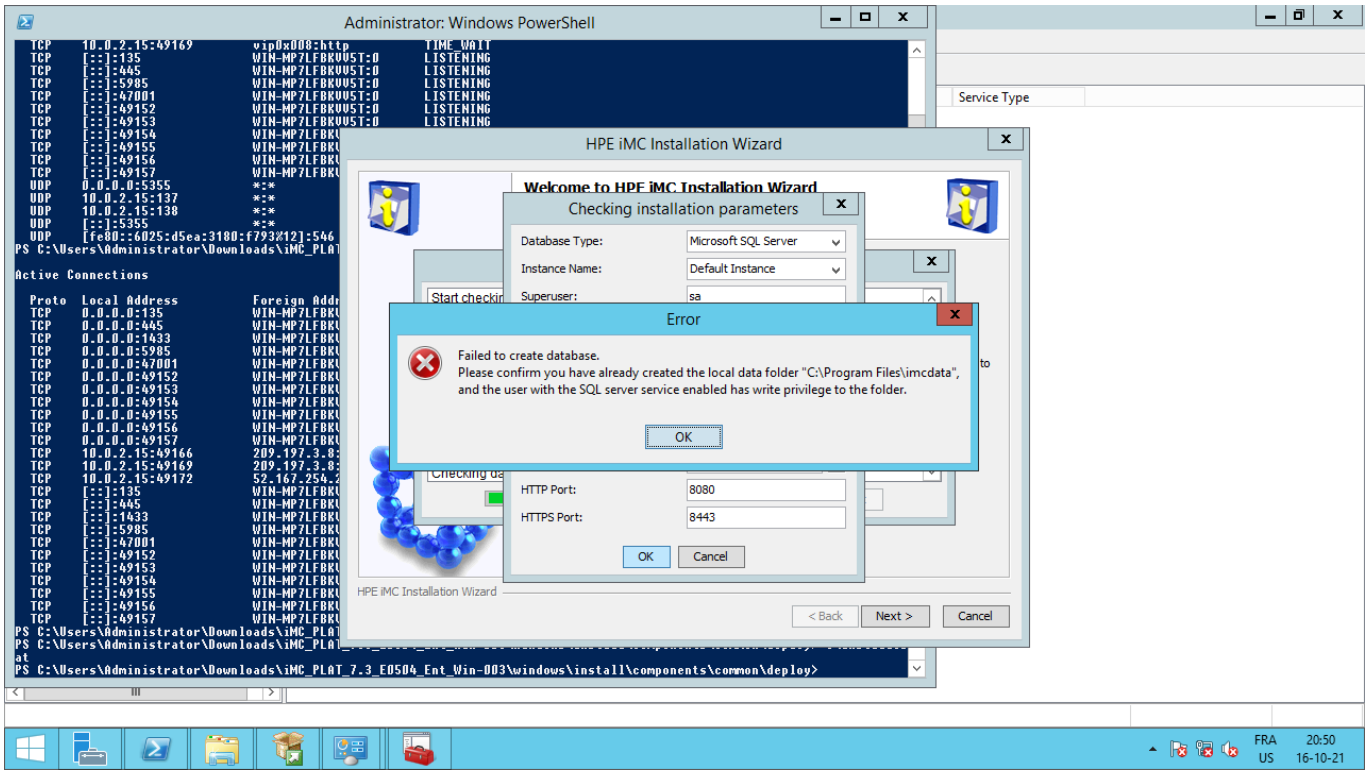

Sql Server Configuration Manager
File Action View Help
SQL Server Configuration Manager (Local)
SQL Server Services
SQL Server Network Configuration (32bit
SQL Native Client 11.0 Configuration (32
SQL Server Network Configuration
Protocols for MSSQLSERVER
SQL Native Client 11.0 Configuration
Protocol Name
Status
Shared Memory
Enabled
Named Pipes
Disabled
TCP/IP
Enabled
FRA
US
10:48
14-10-21


Sql Server Configuration Manager
File Action View Help
SQL Server Configuration Manager (Local)
SQL Server Services
SQL Server Network Configuration (32bit
SQL Native Client 11.0 Configuration (32
SQL Server Network Configuration
Protocols for MSSQLSERVER
SQL Native Client 11.0 Configuration
Protocol Name
Status
Shared Memory
Named Pipes
TCP/IP
TCP/IP Properties
Protocol
IP Addresses
General
Enabled
Yes
Keep Alive
30000
Listen All
Yes
Enabled
Enable or disable TCP/IP protocol for this server instance
OK
Cancel
Apply
Help
FRA
US
18:42
16-10-21

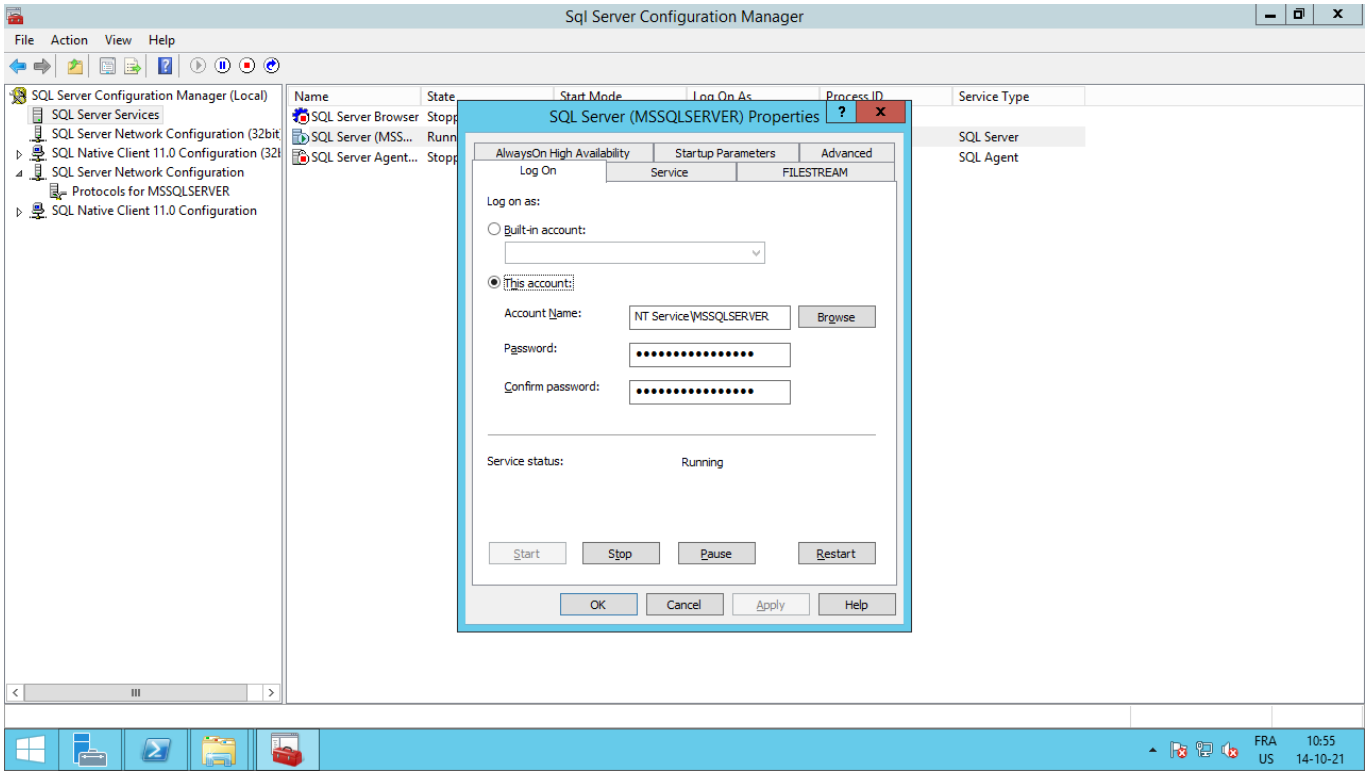

Sql Server Configuration Manager
File Action View Help
SQL Server Configuration Manager (Local)
SQL Server Services
SQL Server Network Configuration (32bit
SQL Native Client 11.0 Configuration (32
SQL Server Network Configuration
Protocols for MSSQLSERVER
SQL Native Client 11.0 Configuration
Name
State
Start Mode
Log On As
Process ID
Service Type
SQL Server Browser
Stopped
Other (Boot, Syste...
NT AUTHORITY\LO...
0
SQL Ser
Automatic
NT Service\MSSQL...
2584
SQL Server
SQL Ser
Other (Boot, Syste...
NT AUTHORITY\NE...
0
SQL Agent
Start
Stop
Pause
Resume
Restart
Properties
Help
Restart (stop and then start) selected service.
FRA
US
10:48
14-10-21

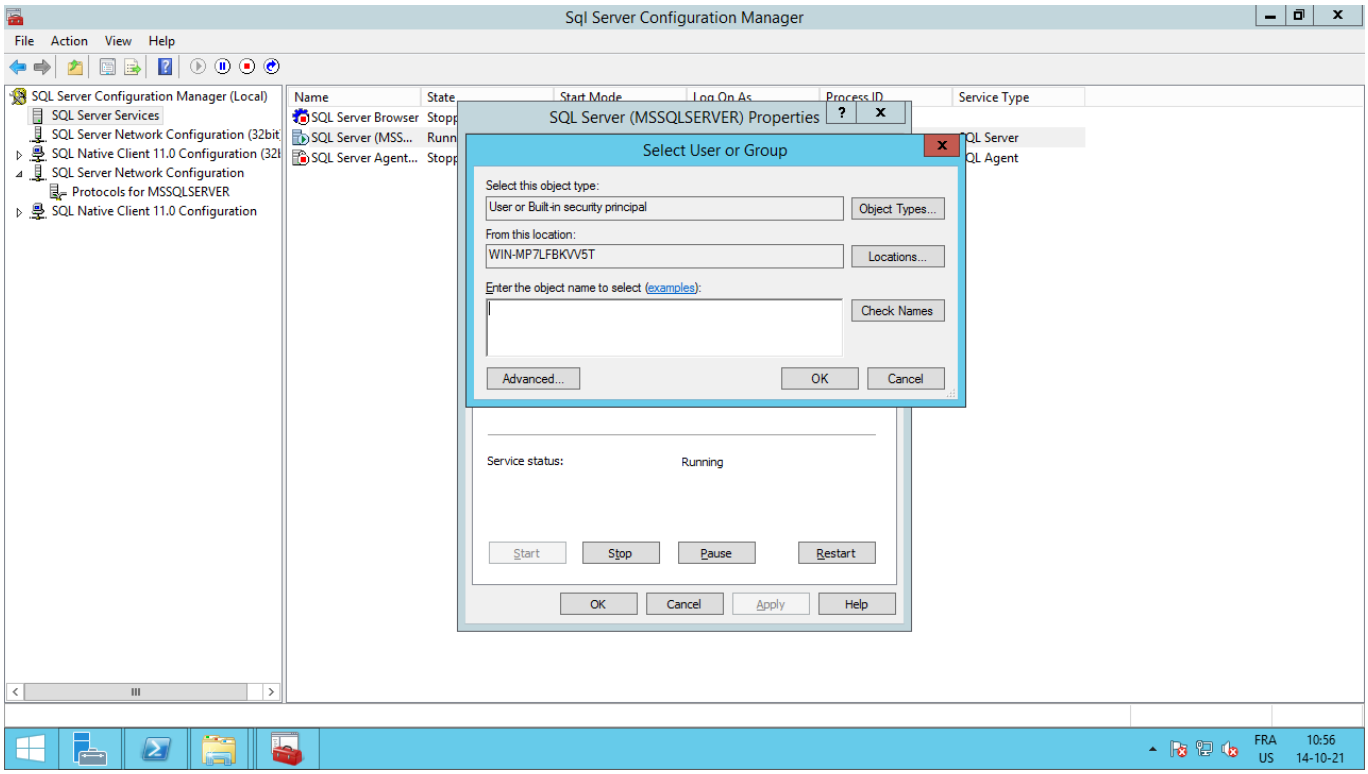


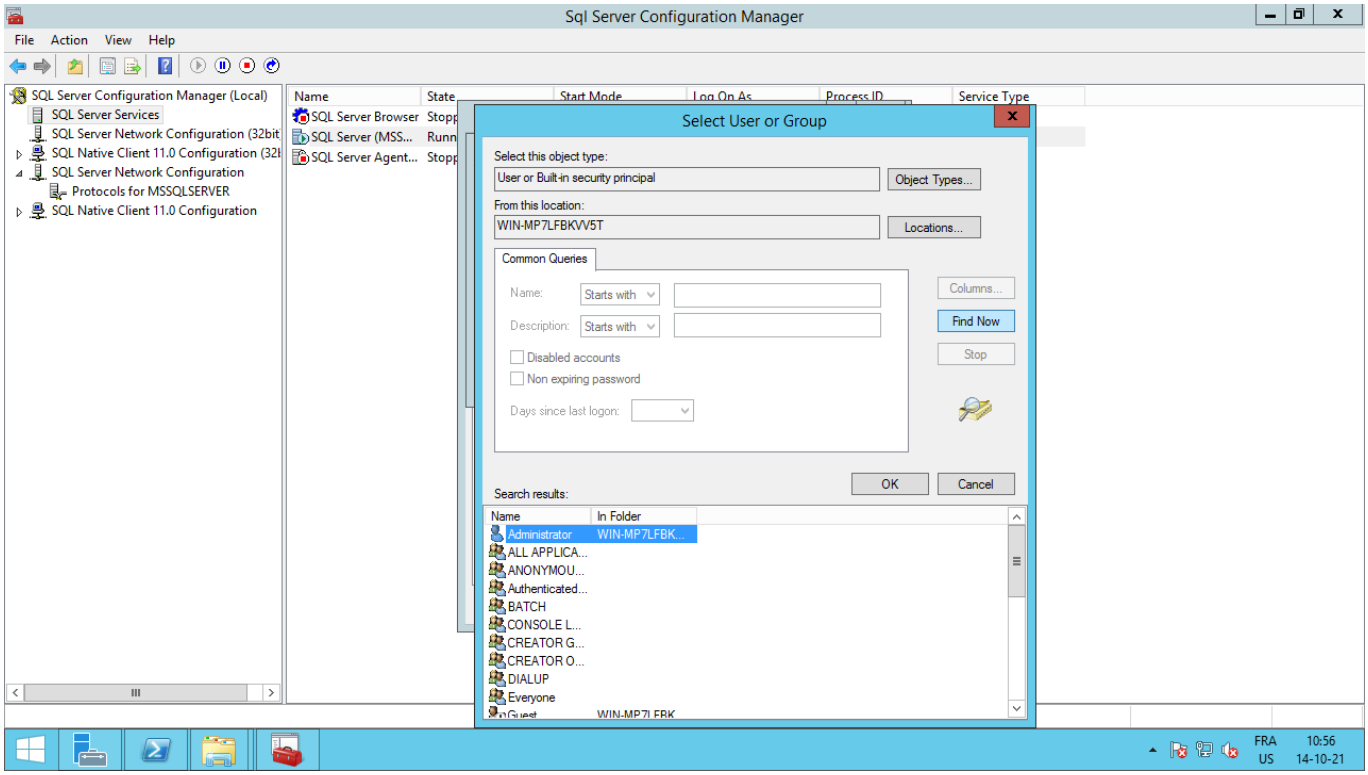
At this point, the SQLServer2014Express is installed and running on port 1433 with default configuration.
However, if the iMC installer is launched it will complain that the current user doesn't have the correct privileges, something like:

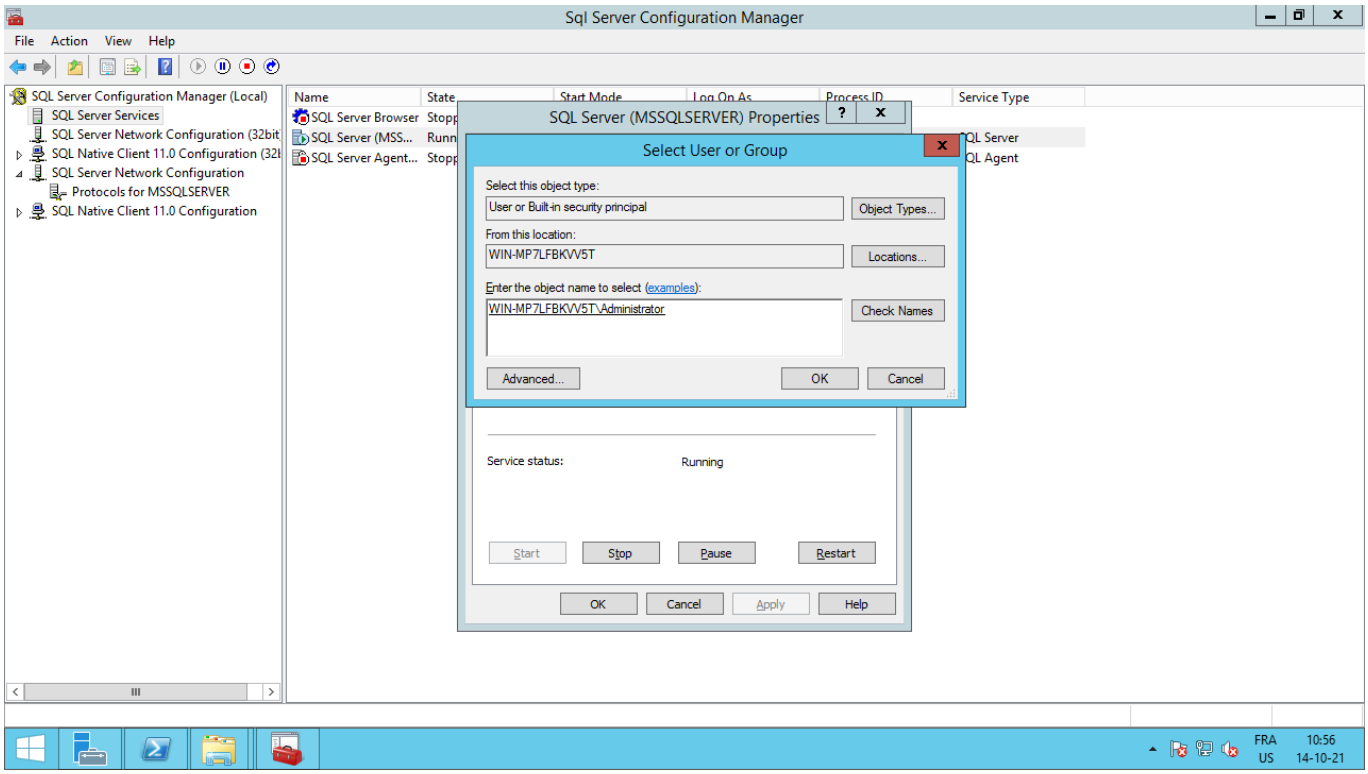


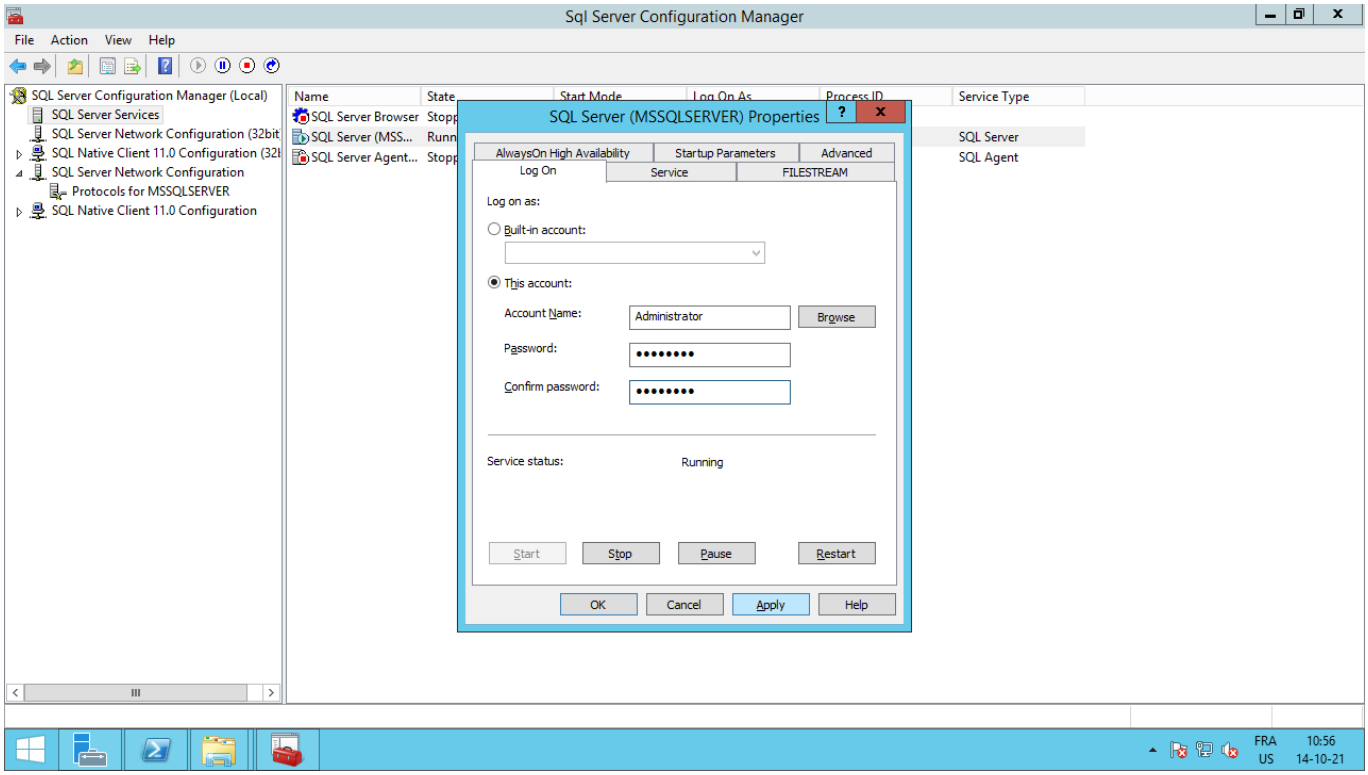
To circumvent this issue, we change the account with which SQLServer is accessed, from the default NT service account to the administrator:

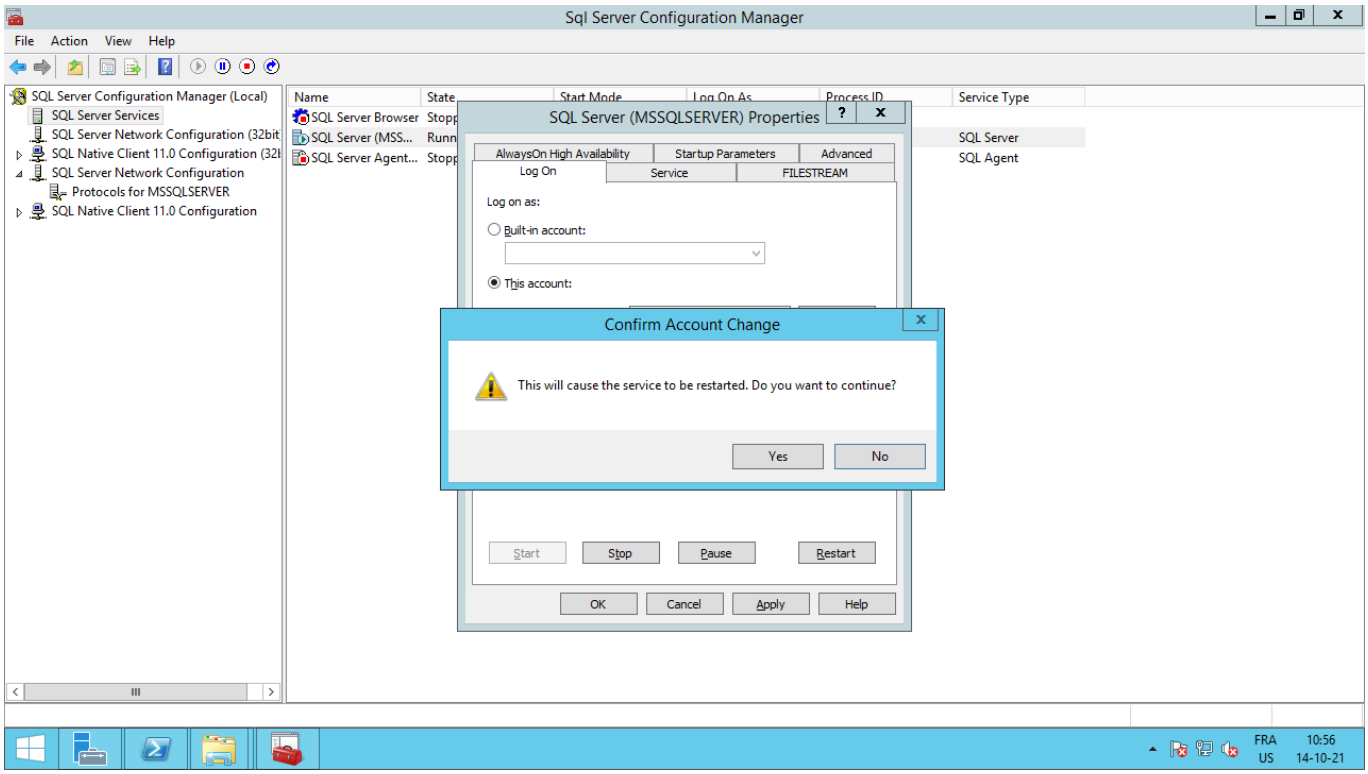

Sql Server Configuration Manager
File Action View Help
SQL Server Configuration Manager (Local)
SQL Server Services
SQL Server Network Configuration (32bit
SQL Native Client 11.0 Configuration (32
SQL Server Network Configuration
Protocols for MSSQLSERVER
SQL Native Client 11.0 Configuration
Name State Start Mode Log On As Process ID Service Type
SQL Server Browser Stopp
SQL Server (MSS... Runn
SQL Server Agent... Stopp
SQL Server
SQL Agent
SQL Server (MSSQLSERVER) Properties
AlwaysOn High Availability
Startup Parameters
Advanced
Log On
Service
FILESTREAM
Log on as:
Built-in account:
This account:
Account Name:
NT Service\MSSQLSERVER
Browse
Password:
Confirm password:
Service status:
Running
Start
Stop
Pause
Restart
OK
Cancel
Apply
Help
FRA
US
10:55
14-10-21


Sql Server Configuration Manager
File Action View Help
SQL Server Configuration Manager (Local)
SQL Server Services
SQL Server Network Configuration (32bit
SQL Native Client 11.0 Configuration (32
SQL Server Network Configuration
Protocols for MSSQLSERVER
SQL Native Client 11.0 Configuration
Name State Start Mode Log On As Process ID Service Type
SQL Server Browser Stopp
SQL Server (MSS... Runn
SQL Server Agent... Stopp
SQL Server (MSSQLSERVER) Properties
Select User or Group
Select this object type:
User or Built-in security principal
Object Types...
From this location:
WIN-MP7LFBKVV5T
Locations...
Enter the object name to select (examples):
Check Names
Advanced...
OK
Cancel
Service status:
Running
Start
Stop
Pause
Restart
OK
Cancel
Apply
Help
FRA
US
10:56
14-10-21


Sql Server Configuration Manager
File Action View Help
SQL Server Configuration Manager (Local)
SQL Server Services
SQL Server Network Configuration (32bit
SQL Native Client 11.0 Configuration (32
SQL Server Network Configuration
Protocols for MSSQLSERVER
SQL Native Client 11.0 Configuration
Name State Start Mode Log On As Process ID Service Type
SQL Server Browser Stopp
SQL Server (MSS... Runn
SQL Server Agent... Stopp
Select User or Group
Select this object type:
User or Built-in security principal
Object Types...
From this location:
WIN-MP7LFBKVV5T
Locations...
Common Queries
Name: Starts with
Description: Starts with
Disabled accounts
Non expiring password
Days since last logon:
Columns...
Find Now
Stop
OK
Cancel
Search results:
Name In Folder
Administrator WIN-MP7LFBK...
ALL APPLICA...
ANONYMOU...
Authenticated...
BATCH
CONSOLE L...
CREATOR G...
CREATOR O...
DIALUP
Everyone
FRA
US
10:56
14-10-21


Sql Server Configuration Manager
File Action View Help
SQL Server Configuration Manager (Local)
SQL Server Services
SQL Server Network Configuration (32bit
SQL Native Client 11.0 Configuration (32
SQL Server Network Configuration
Protocols for MSSQLSERVER
SQL Native Client 11.0 Configuration
Name State Start Mode Log On As Process ID Service Type
SQL Server Browser Stopp
SQL Server (MSS... Runn
SQL Server Agent... Stopp
SQL Server (MSSQLSERVER) Properties
Select User or Group
Select this object type:
User or Built-in security principal
Object Types...
From this location:
WIN-MP7LFBKVV5T
Locations...
Enter the object name to select (examples):
WIN-MP7LFBKVV5T\Administrator
Check Names
Advanced...
OK
Cancel
Service status:
Running
Start
Stop
Pause
Restart
OK
Cancel
Apply
Help
FRA
US
10:56
14-10-21





Given this configuration, we should be able to move flawlessly through the iMC installer.

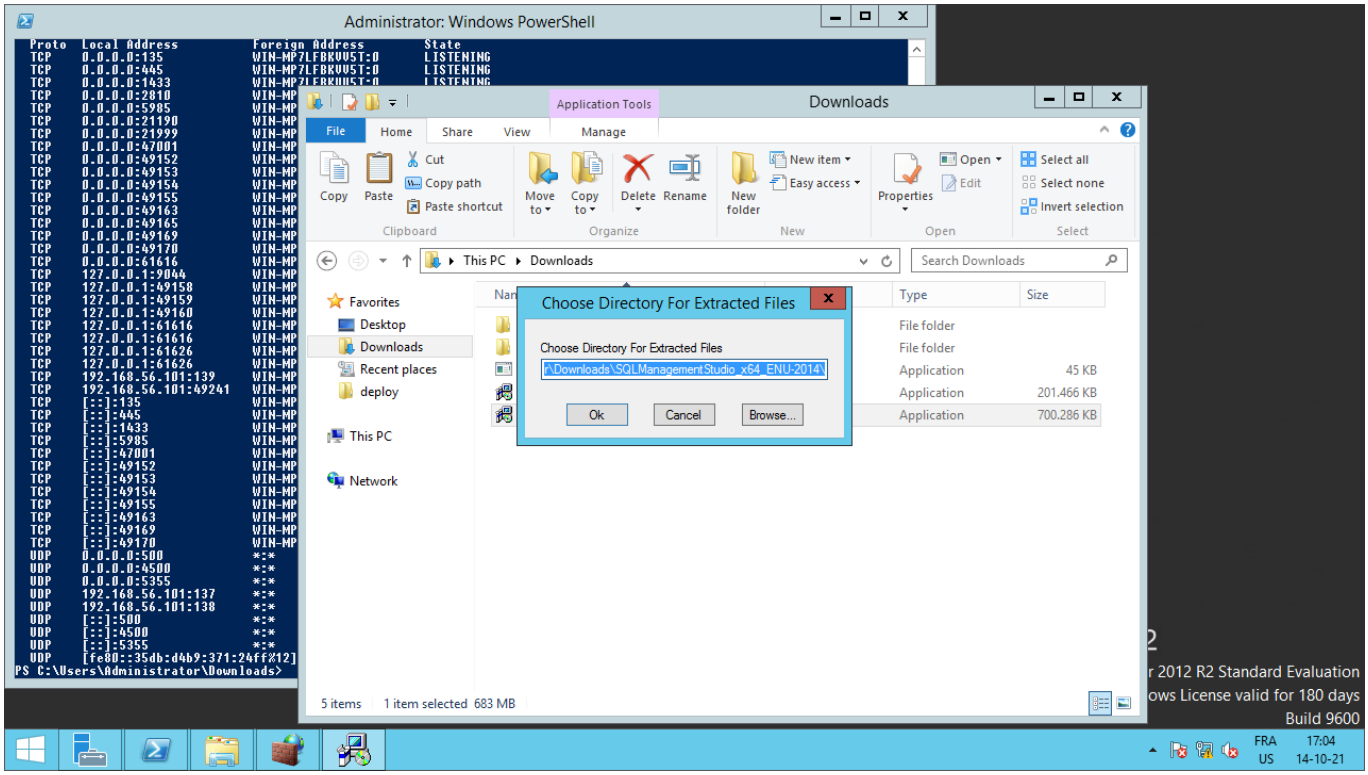
SQLManagementStudio2014:

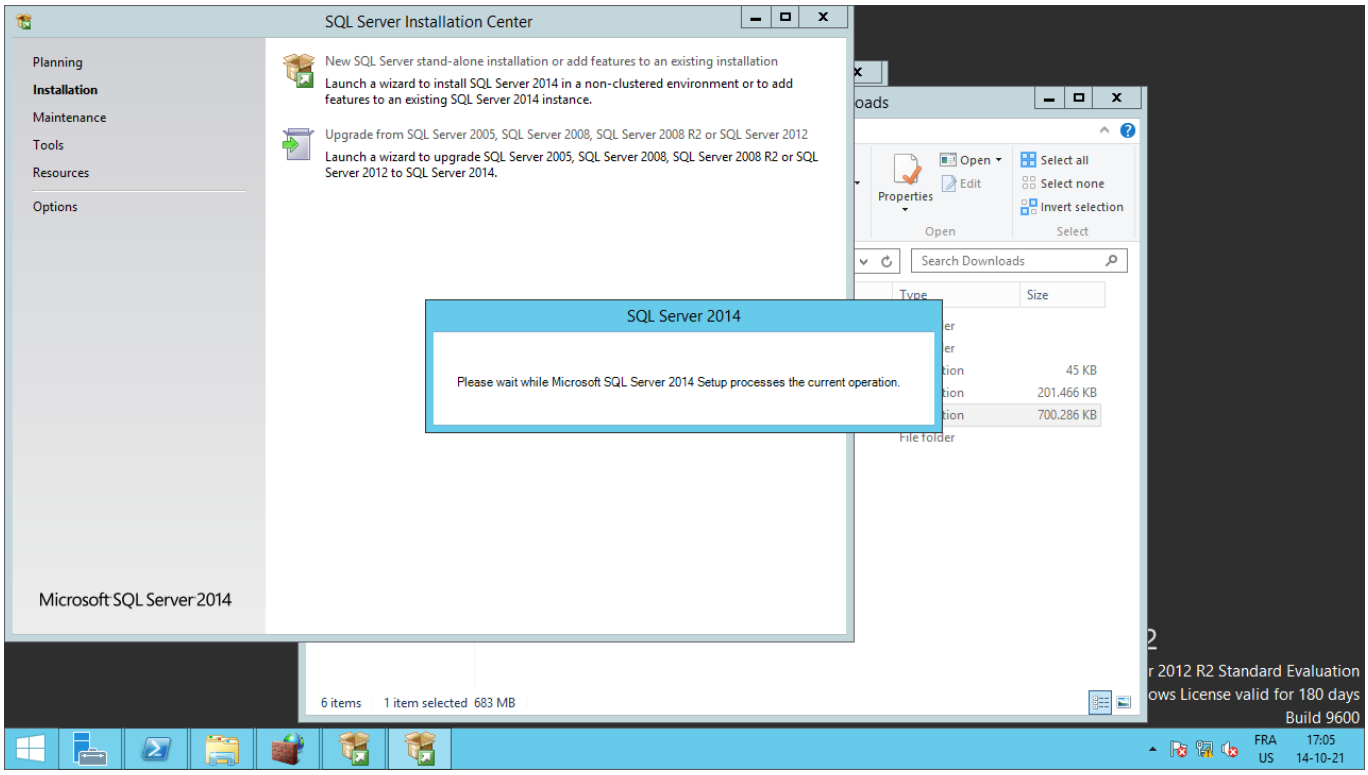
```
$ sha1sum SQLManagementStudio_x64_ENU-2014.exe
406a6dd55552d35fd99229c4d13e43375f131828  SQLManagementStudio_x64_ENU-2014.exe
```

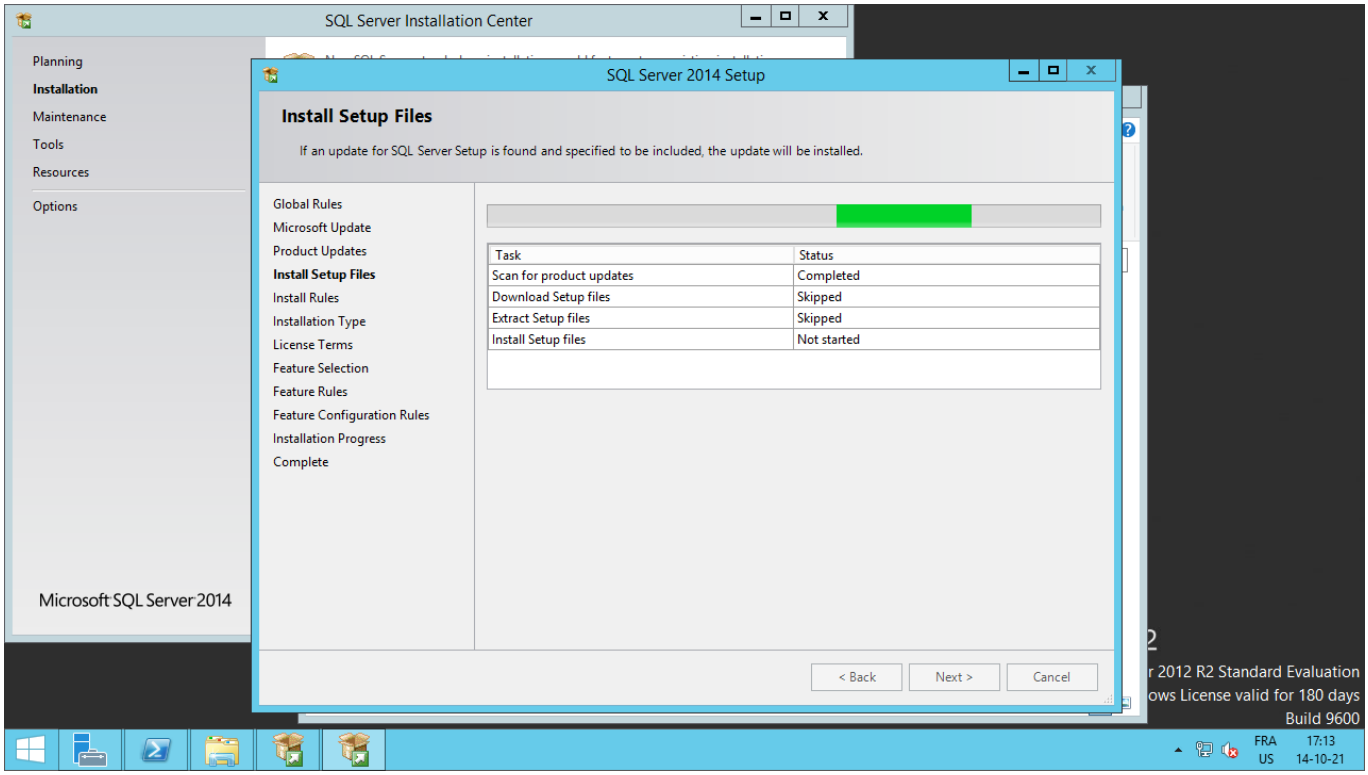
https://www.microsoft.com/en-us/download/details.aspx?id=42299

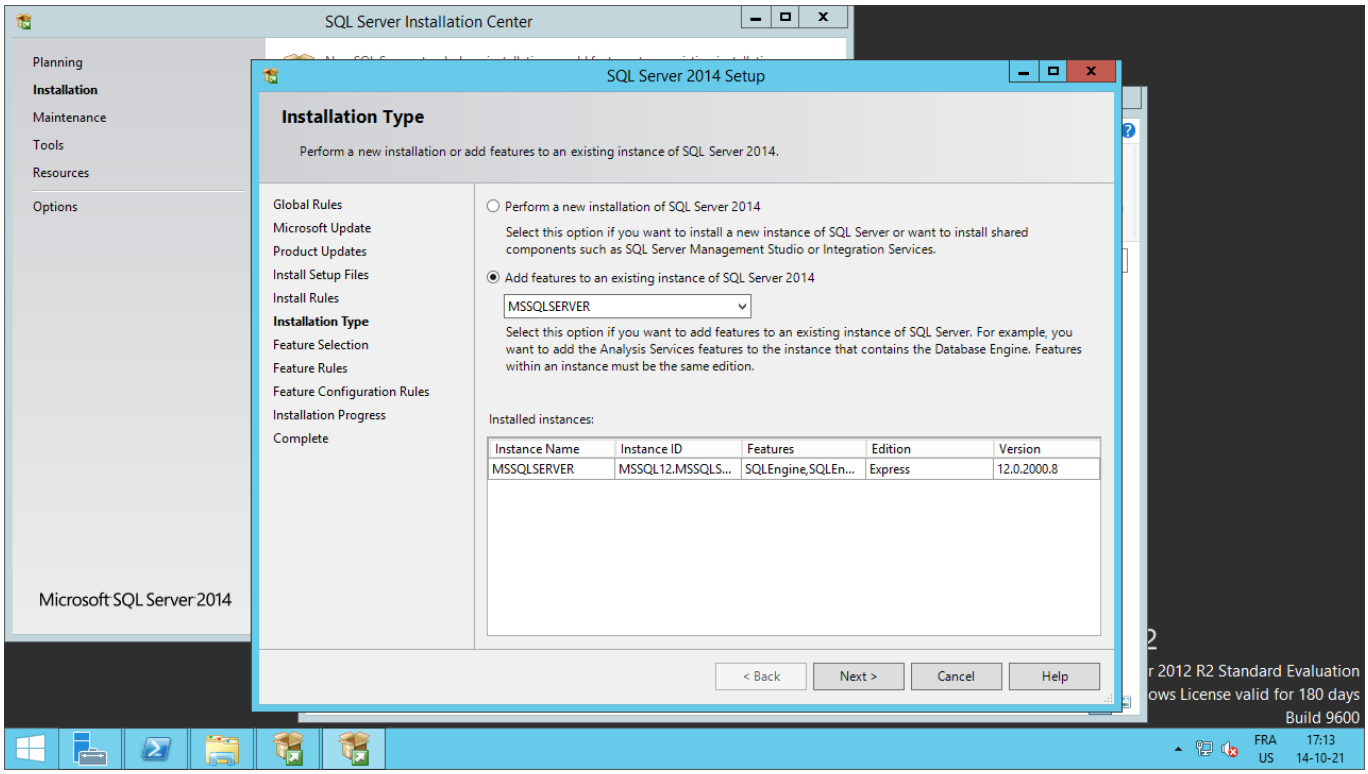
SQLManagementStudio is actually not needed to run iMC but gives the ability to explore the *sa* account.

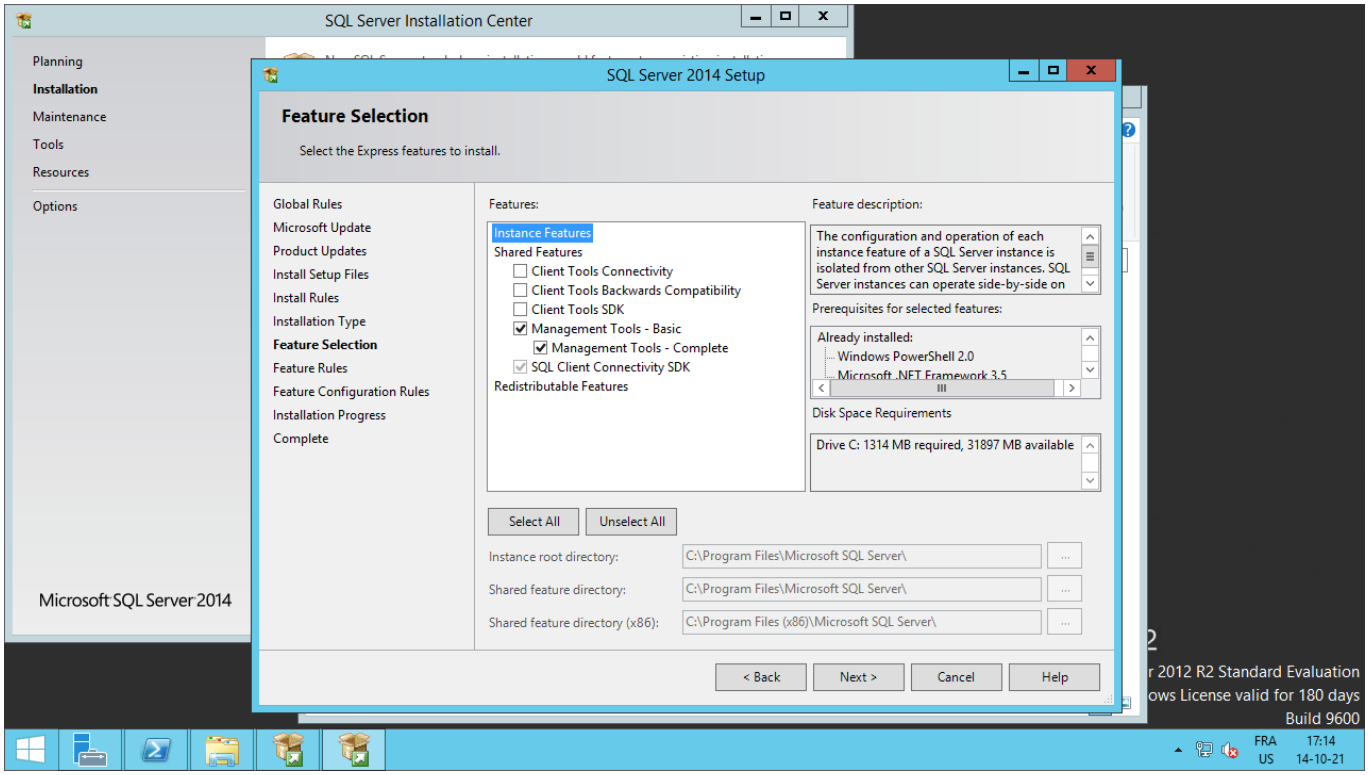
Installation steps:

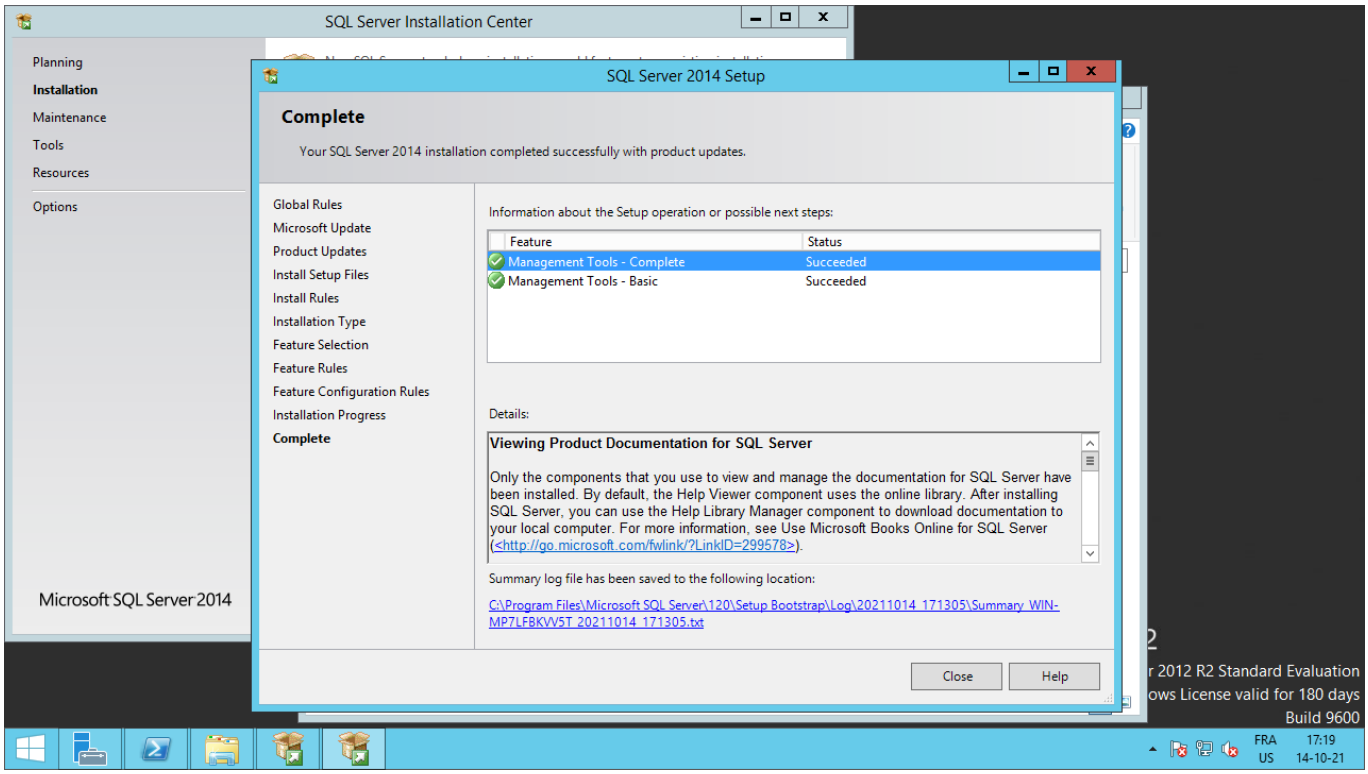


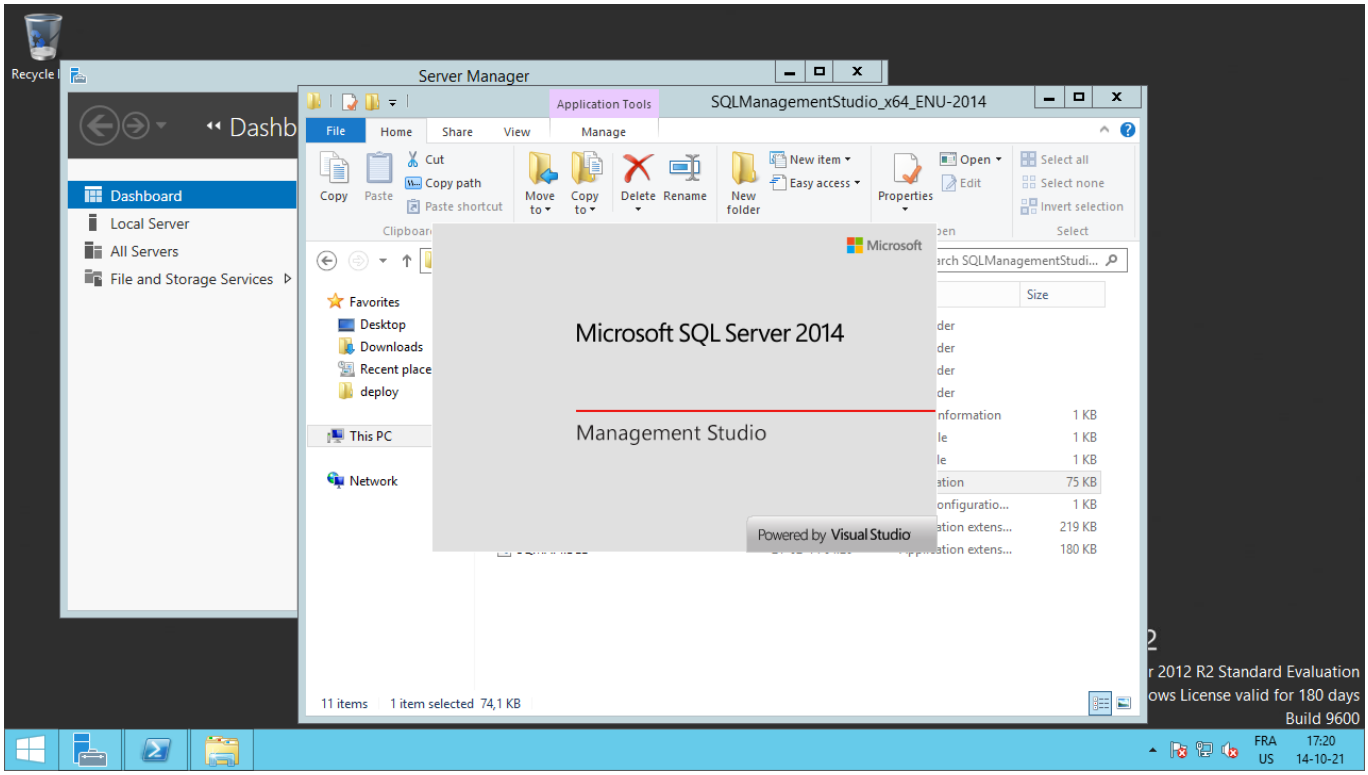


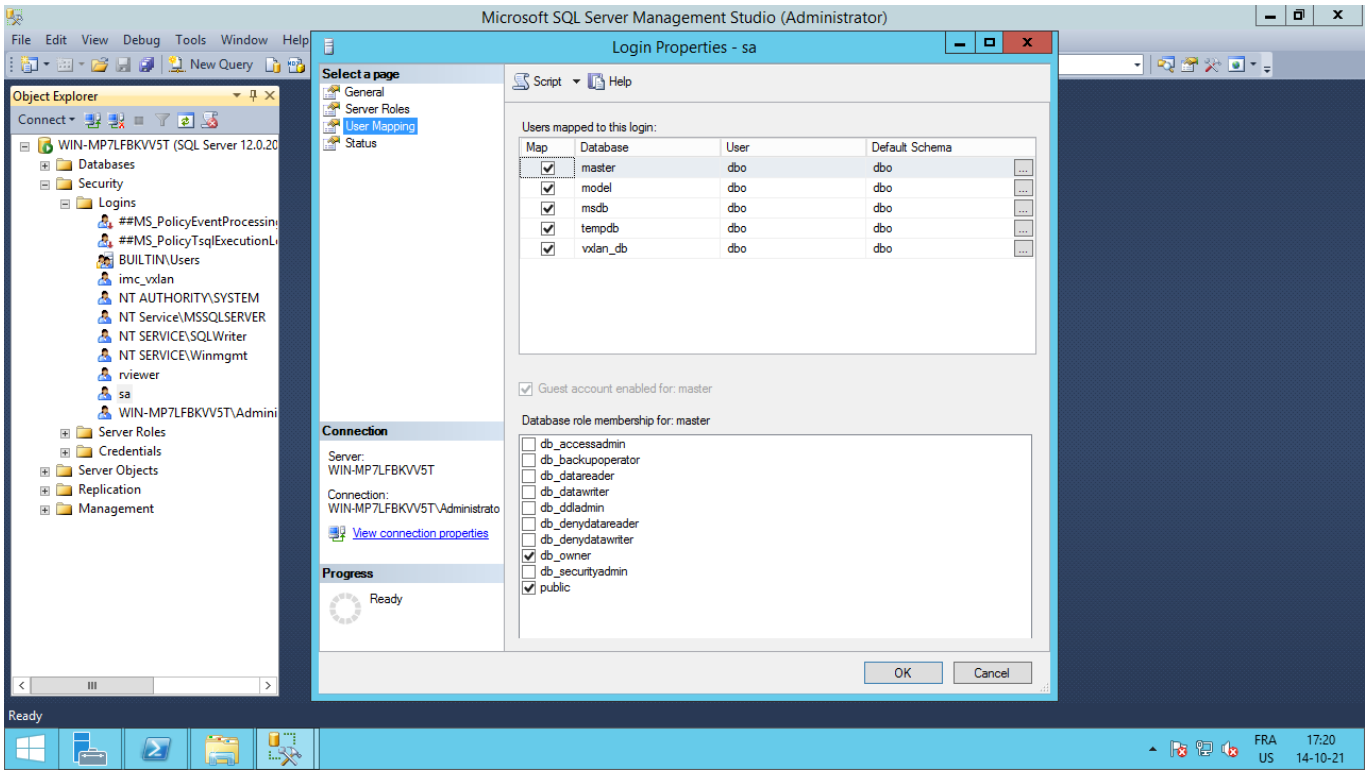

SQL Server Installation Center
Planning
Installation
Maintenance
Tools
Resources
Options
Microsoft SQL Server 2014
SQL Server 2014 Setup
Install Setup Files
If an update for SQL Server Setup is found and specified to be included, the update will be installed.
Global Rules
Microsoft Update
Product Updates
Install Setup Files
Install Rules
Installation Type
License Terms
Feature Selection
Feature Rules
Feature Configuration Rules
Installation Progress
Complete
Task
Status
Scan for product updates
Completed
Download Setup files
Skipped
Extract Setup files
Skipped
Install Setup files
Not started
< Back
Next >
Cancel
r 2012 R2 Standard Evaluation
ows License valid for 180 days
Build 9600
FRA
US
17:13
14-10-21


SQL Server Installation Center
Planning
Installation
Maintenance
Tools
Resources
Options
Microsoft SQL Server 2014
SQL Server 2014 Setup
Installation Type
Perform a new installation or add features to an existing instance of SQL Server 2014.
Global Rules
Microsoft Update
Product Updates
Install Setup Files
Install Rules
Installation Type
Feature Selection
Feature Rules
Feature Configuration Rules
Installation Progress
Complete
Perform a new installation of SQL Server 2014
Select this option if you want to install a new instance of SQL Server or want to install shared components such as SQL Server Management Studio or Integration Services.
Add features to an existing instance of SQL Server 2014
MSSQLSERVER
Select this option if you want to add features to an existing instance of SQL Server. For example, you want to add the Analysis Services features to the instance that contains the Database Engine. Features within an instance must be the same edition.
Installed instances:
Instance Name
Instance ID
Features
Edition
Version
MSSQLSERVER
MSSQL12.MSSQLS...
SQLEngine,SQLEn...
Express
12.0.2000.8
< Back
Next >
Cancel
Help
r 2012 R2 Standard Evaluation
ows License valid for 180 days
Build 9600
FRA
US
17:13
14-10-21


SQL Server Installation Center
Planning
Installation
Maintenance
Tools
Resources
Options
Microsoft SQL Server 2014
SQL Server 2014 Setup
Feature Selection
Select the Express features to install.
Global Rules
Microsoft Update
Product Updates
Install Setup Files
Install Rules
Installation Type
Feature Selection
Feature Rules
Feature Configuration Rules
Installation Progress
Complete
Features:
Instance Features
Shared Features
Client Tools Connectivity
Client Tools Backwards Compatibility
Client Tools SDK
Management Tools - Basic
Management Tools - Complete
SQL Client Connectivity SDK
Redistributable Features
Feature description:
The configuration and operation of each instance feature of a SQL Server instance is isolated from other SQL Server instances. SQL Server instances can operate side-by-side on
Prerequisites for selected features:
Already installed:
Windows PowerShell 2.0
Microsoft .NET Framework 3.5
Disk Space Requirements
Drive C: 1314 MB required, 31897 MB available
Select All
Unselect All
Instance root directory:
C:\Program Files\Microsoft SQL Server\
Shared feature directory:
C:\Program Files\Microsoft SQL Server\
Shared feature directory (x86):
C:\Program Files (x86)\Microsoft SQL Server\
< Back
Next >
Cancel
Help
r 2012 R2 Standard Evaluation
ows License valid for 180 days
Build 9600
FRA
US
17:14
14-10-21


SQL Server Installation Center
Planning
Installation
Maintenance
Tools
Resources
Options
Microsoft SQL Server 2014
SQL Server 2014 Setup
Complete
Your SQL Server 2014 installation completed successfully with product updates.
Global Rules
Microsoft Update
Product Updates
Install Setup Files
Install Rules
Installation Type
Feature Selection
Feature Rules
Feature Configuration Rules
Installation Progress
Complete
Information about the Setup operation or possible next steps:
Feature
Status
Management Tools - Complete
Succeeded
Management Tools - Basic
Succeeded
Details:
Viewing Product Documentation for SQL Server
Only the components that you use to view and manage the documentation for SQL Server have been installed. By default, the Help Viewer component uses the online library. After installing SQL Server, you can use the Help Library Manager component to download documentation to your local computer. For more information, see Use Microsoft Books Online for SQL Server (<http://go.microsoft.com/fwlink/?LinkID=299578>).
Summary log file has been saved to the following location:
C:\Program Files\Microsoft SQL Server\120\Setup Bootstrap\Log\20211014_171305\Summary_WIN-MP7LFBKVV5T_20211014_171305.txt
Close
Help
r 2012 R2 Standard Evaluation
ows License valid for 180 days
Build 9600
FRA
US
17:19
14-10-21





iMC_PLAT_7.3_E0504_Ent_Win-003(vuln) or iMC_PLAT_7.3_E0504P04_Win-002(patched)

The installation of iMC is identical for either version. However, there are discrepancies between the content of both folder/archive.

There at least a folder(named db) that is present in:

```
iMC_PLAT_7.3_E0504_Ent_Win-003/
```

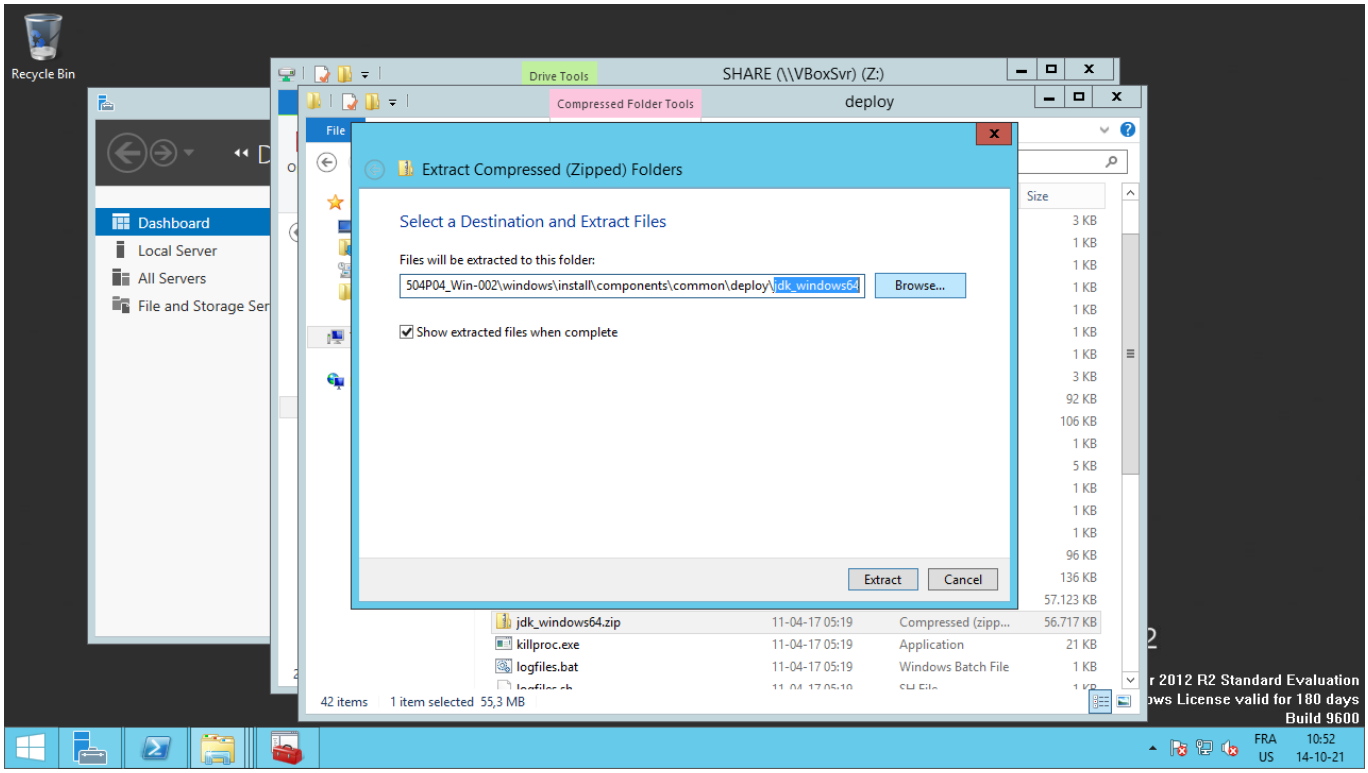
which isn't in the other. This is also the case for at least tool/utility like "envcheck.zip".

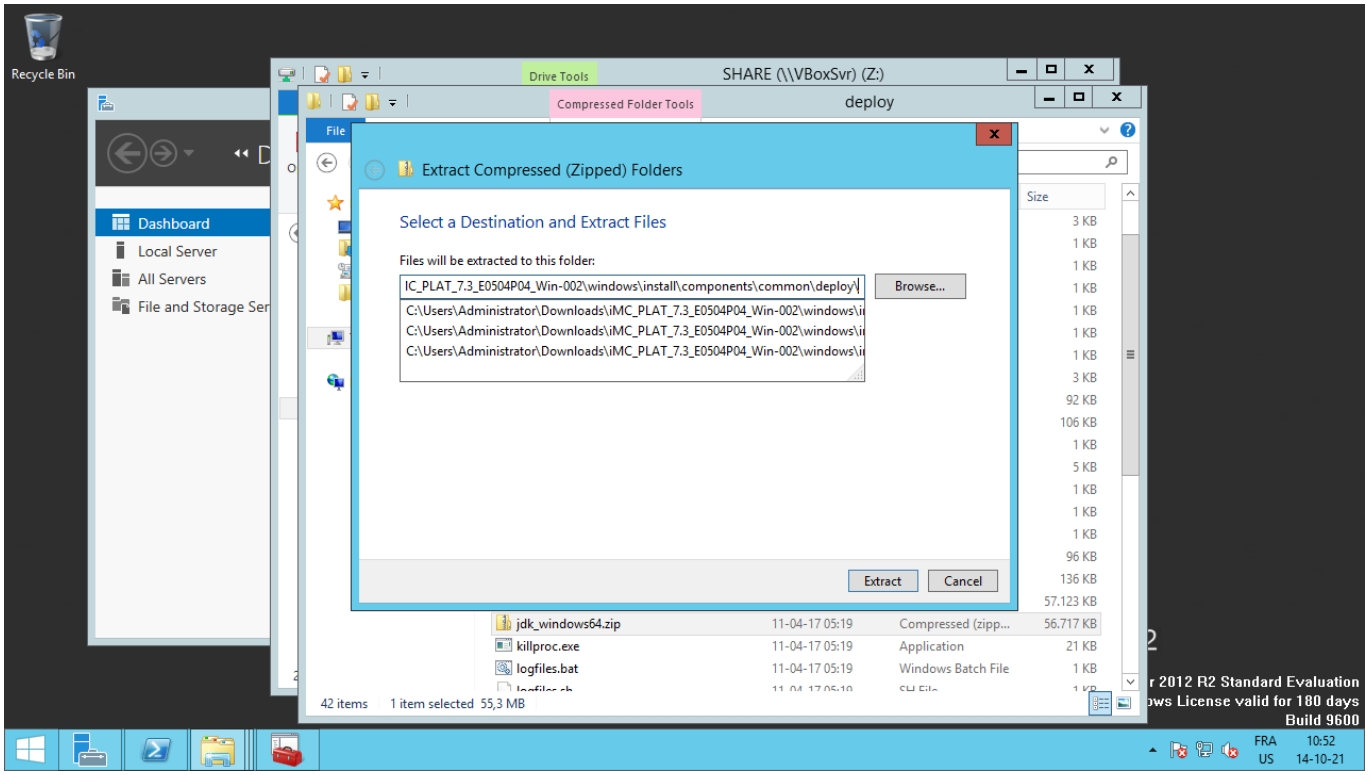
```
ixidor@pr0c:~/exp/iMC_PLAT_7.3_E0504_Ent_Win-003/windows/install/components/common$ ls -l ; tree db
total 16
drwxrwxr-x 2 ixidor ixidor 4096 Jan 23  2017 client
drwxrwxr-x 2 ixidor ixidor 4096 Jan 23  2017 db
drwxrwxr-x 3 ixidor ixidor 4096 Jan 23  2017 deploy
drwxrwxr-x 2 ixidor ixidor 4096 Jan 23  2017 server
db
├── install32.bat
├── install64.bat
├── SQLEXPR_x64_ENU.exe
├── SQLEXPR_x86_ENU.exe
├── SQLServer2008R2-KB3045313-x64.exe
├── SQLServer2008R2-KB3045313-x86.exe
├── uninstall32.bat
├── uninstall64.bat
├── update32.bat
└── update64.bat

0 directories, 10 files
ixidor@pr0c:~/exp/iMC_PLAT_7.3_E0504_Ent_Win-003/windows/install/components/common$ ls ../../../tools/
cluster_tools.zip    dhcp-plug-windows.zip  iHATool.zip                 iMC_OVM.zip    InstallGO              lldp-agent-ubuntu.zip   MSKBsC              SCOM.zip  vrm-plug-linux.zip
dhcp-plug-linux.zip  envcheck.zip           iMC-MIB-Download_Windows.zip  imc_tools.exe  lldp-agent-redhat.zip  lldp-agent-windows.zip  openstack-plug.zip  VPD       vrm-plug-windows.zip
ixidor@pr0c:~/exp/iMC_PLAT_7.3_E0504_Ent_Win-003/windows/install/components/common$


ixidor@pr0c:~/exp/iMC_PLAT_7.3_E0504P04_Win-002/windows/install/components/common$ ls -l
total 12
drwxrwxr-x 2 ixidor ixidor 4096 Mai  2  2017 client
drwxrwxr-x 3 ixidor ixidor 4096 Mai  2  2017 deploy
drwxrwxr-x 2 ixidor ixidor 4096 Mai  2  2017 server
ixidor@pr0c:~/exp/iMC_PLAT_7.3_E0504P04_Win-002/windows/install/components/common$ ls ../../../tools/
dhcp-plug-linux.zip    iHATool.zip                 iMC-MIB-Download_Windows.zip  lldp-agent-ubuntu.zip   MSKBsC.zip          vrm-plug-linux.zip
dhcp-plug-windows.zip  iMC_MIB_Download_Linux.zip  lldp-agent-redhat.zip         lldp-agent-windows.zip  openstack-plug.zip  vrm-plug-windows.zip
ixidor@pr0c:~/exp/iMC_PLAT_7.3_E0504P04_Win-002/windows/install/components/common$
```

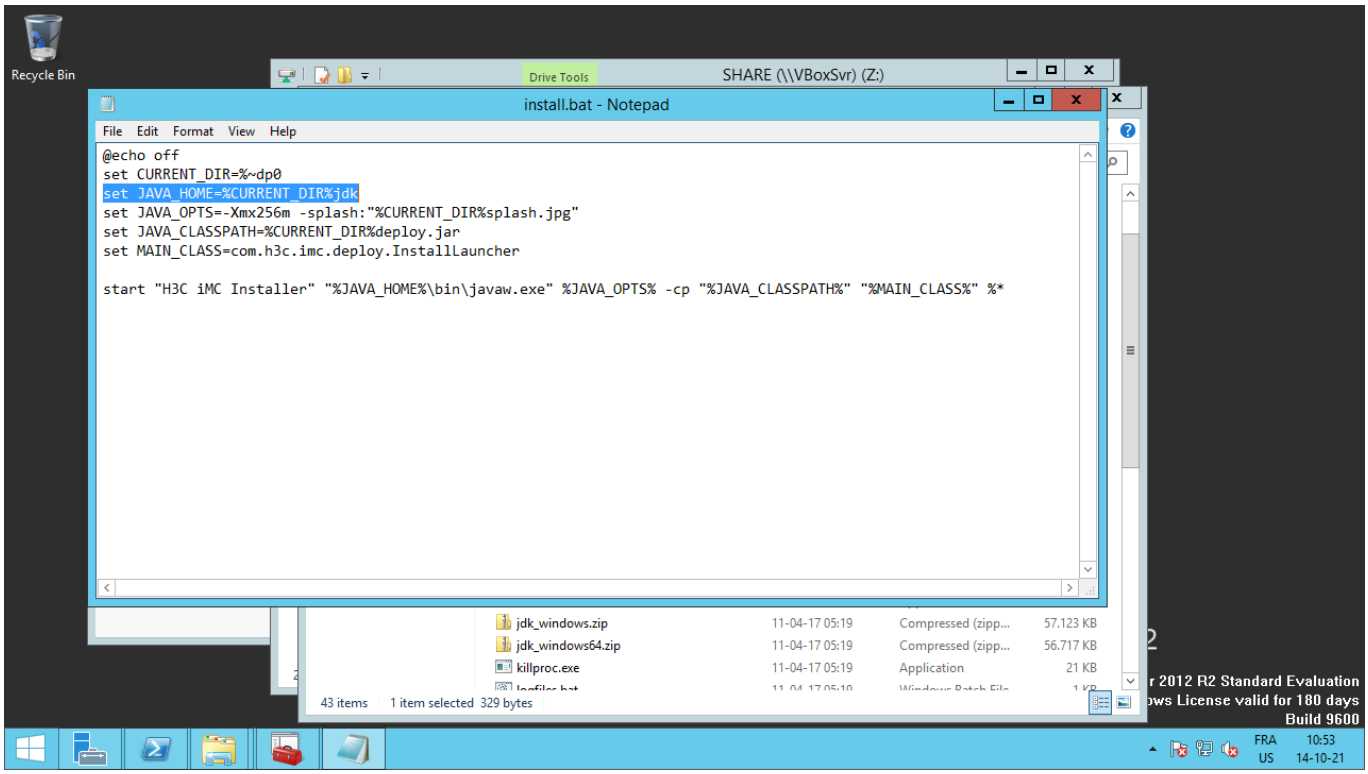
Installation steps:

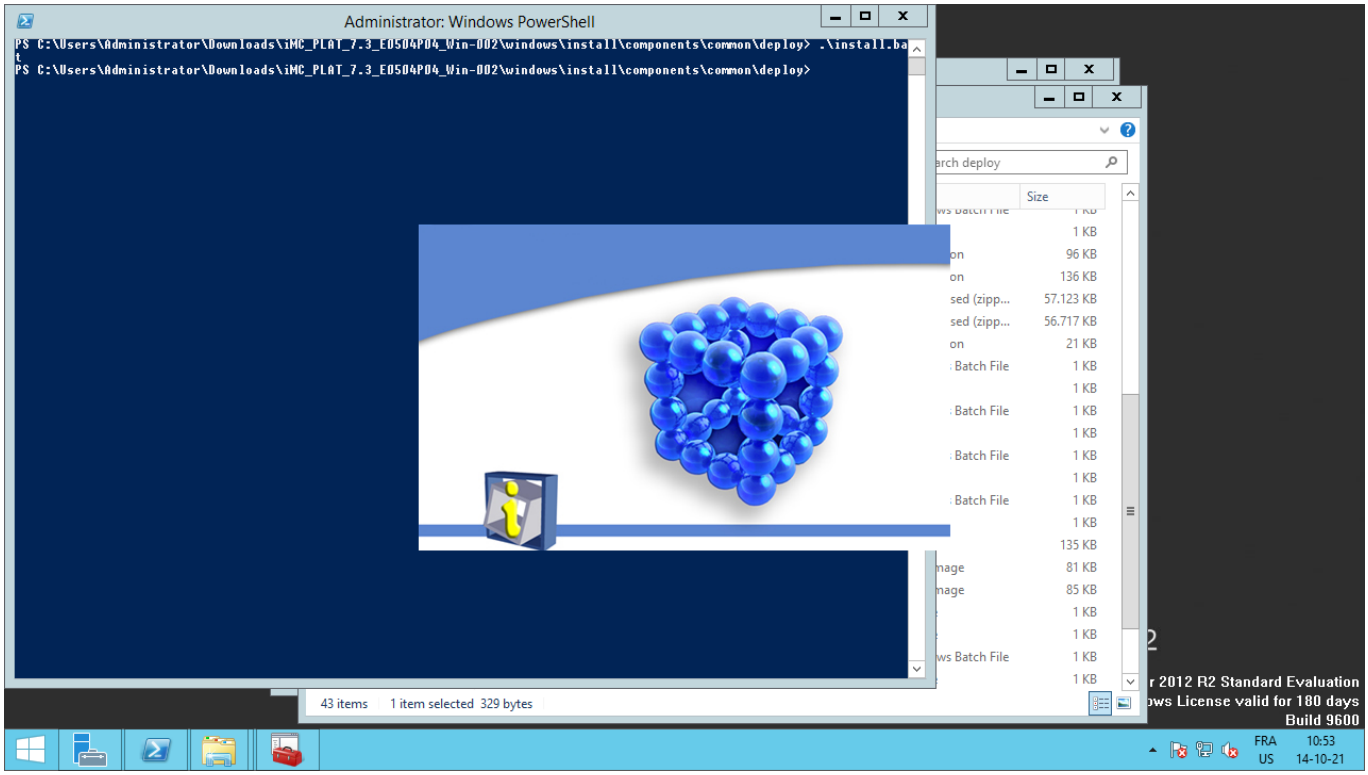
If the previous steps have been followed, there shouldn't be any issues with this point. The only requirement is to unzip and place the JDK:

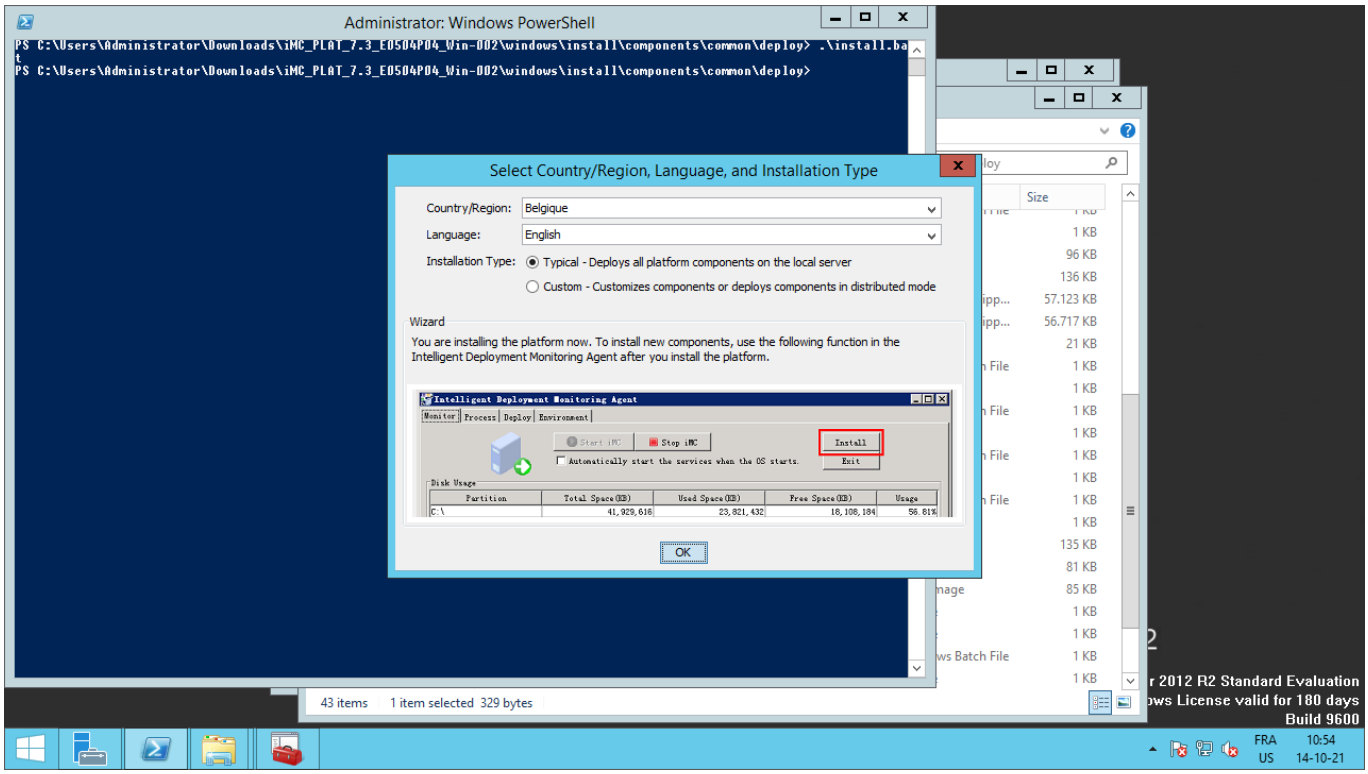
```
C:\Users\Administrator\Downloads\iMC_PLAT_7.3_E0504P04_Win-002\windows\install\components\common\deploy
```

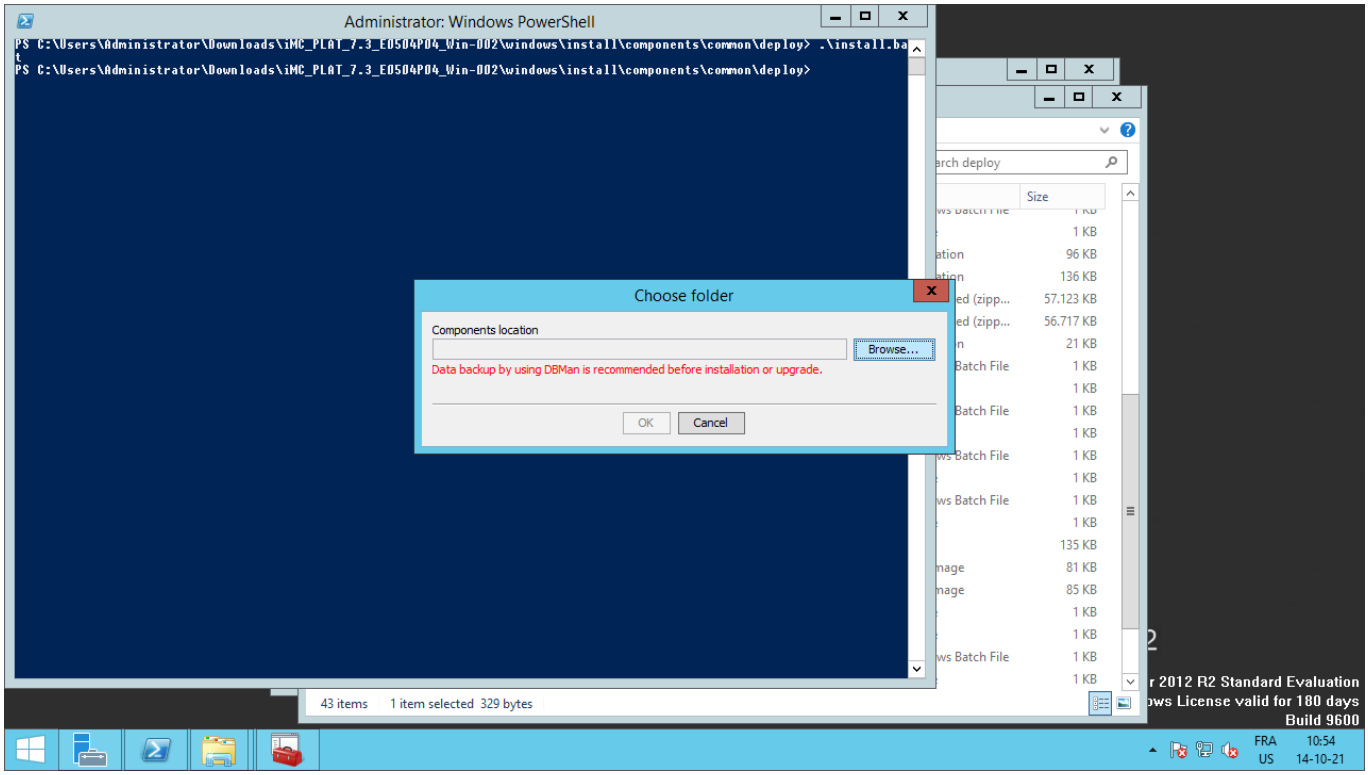


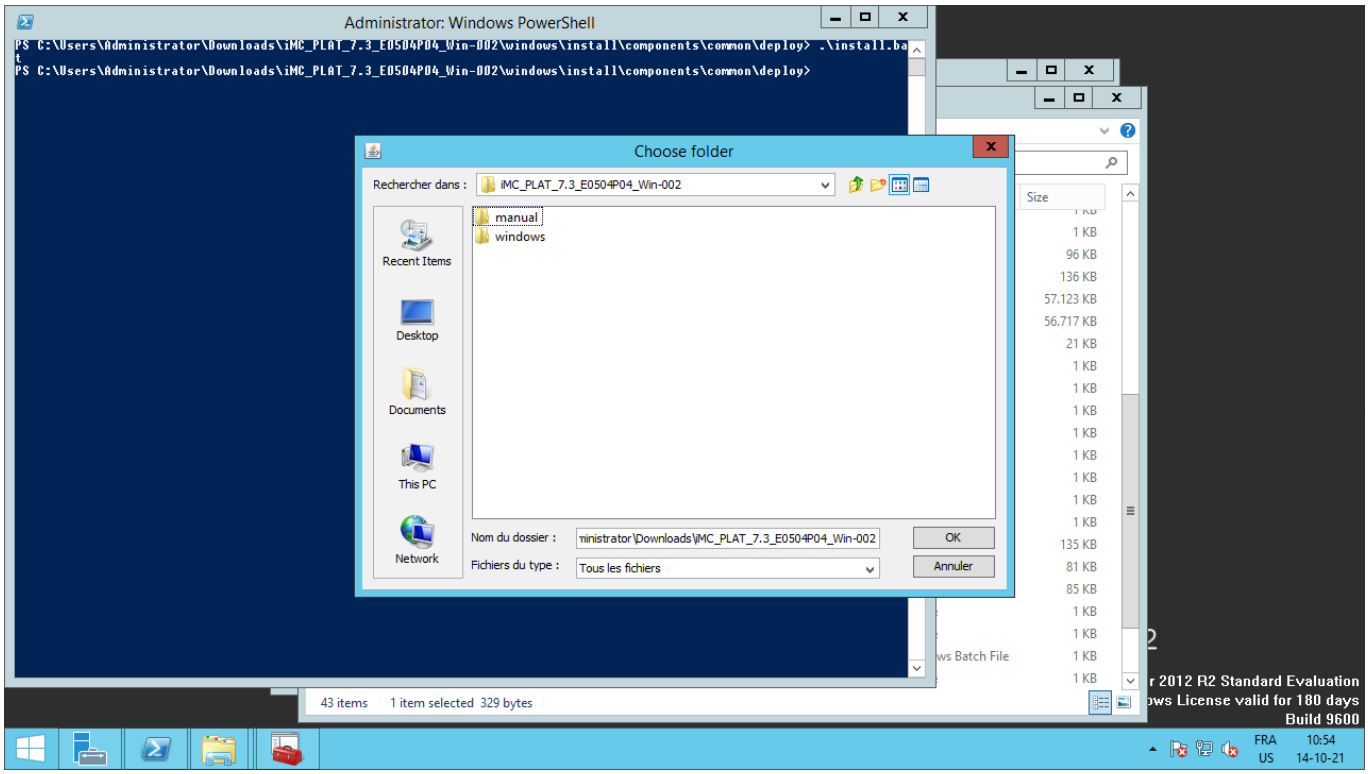

Recycle Bin
Drive Tools
SHARE (\\VBoxSvr) (Z:)
Compressed Folder Tools
deploy
File
Extract Compressed (Zipped) Folders
Select a Destination and Extract Files
Files will be extracted to this folder:
IC_PLAT_7.3_E0504P04_Win-002\windows\install\components\common\deploy\
Browse...
C:\Users\Administrator\Downloads\iMC_PLAT_7.3_E0504P04_Win-002\windows\i
C:\Users\Administrator\Downloads\iMC_PLAT_7.3_E0504P04_Win-002\windows\i
C:\Users\Administrator\Downloads\iMC_PLAT_7.3_E0504P04_Win-002\windows\i
Extract
Cancel
Dashboard
Local Server
All Servers
File and Storage Ser
Size
jdk_windows64.zip 11-04-17 05:19 Compressed (zipp... 56.717 KB
killproc.exe 11-04-17 05:19 Application 21 KB
logfiles.bat 11-04-17 05:19 Windows Batch File 1 KB
42 items 1 item selected 55,3 MB
r 2012 R2 Standard Evaluation
ows License valid for 180 days
Build 9600
FRA
US
10:52
14-10-21

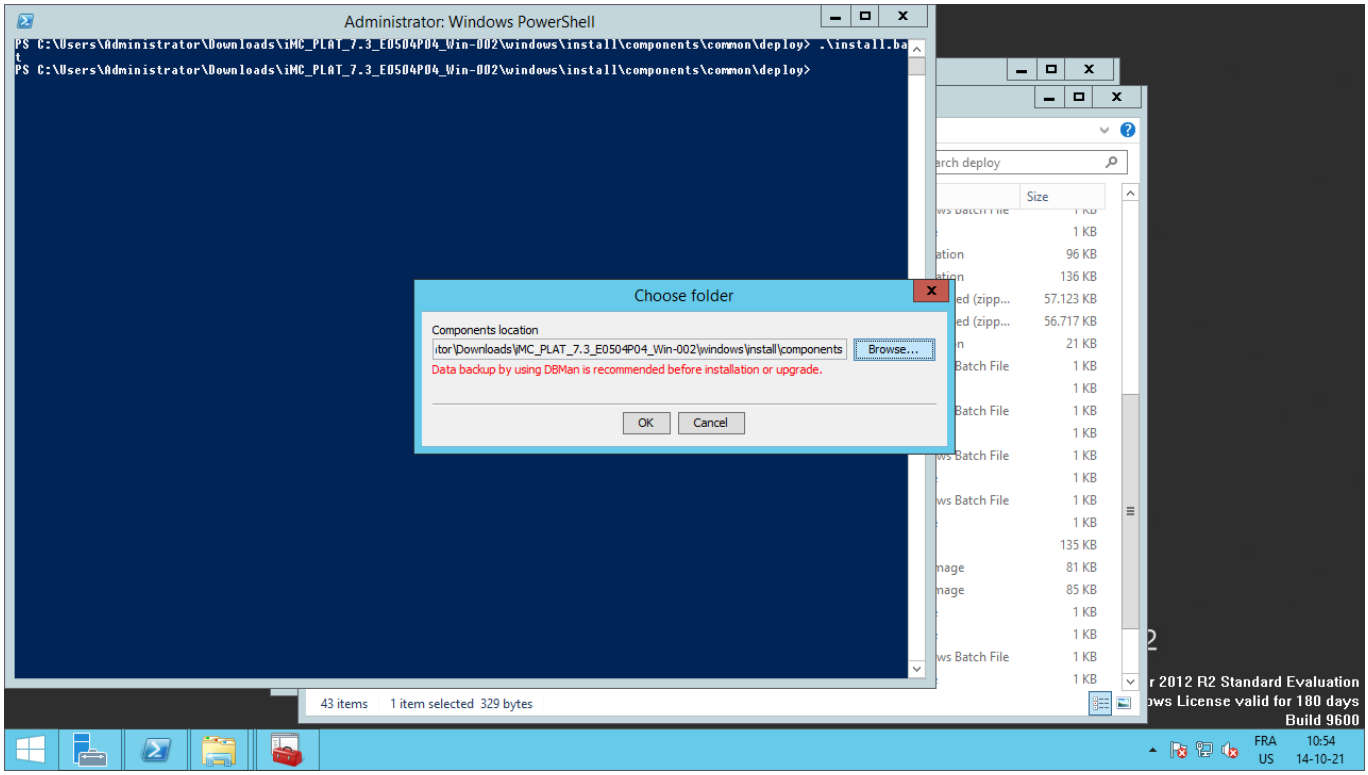

Recycle Bin
Drive Tools
SHARE (\\VBoxSvr) (Z:)
install.bat - Notepad
File Edit Format View Help
@echo off
set CURRENT_DIR=%~dp0
set JAVA_HOME=%CURRENT_DIR%jdk
set JAVA_OPTS=-Xmx256m -splash:"%CURRENT_DIR%splash.jpg"
set JAVA_CLASSPATH=%CURRENT_DIR%deploy.jar
set MAIN_CLASS=com.h3c.imc.deploy.InstallLauncher
start "H3C iMC Installer" "%JAVA_HOME%\bin\javaw.exe" %JAVA_OPTS% -cp "%JAVA_CLASSPATH%" "%MAIN_CLASS%" %*
jdk_windows.zip 11-04-17 05:19 Compressed (zipp... 57.123 KB
jdk_windows64.zip 11-04-17 05:19 Compressed (zipp... 56.717 KB
killproc.exe 11-04-17 05:19 Application 21 KB
43 items 1 item selected 329 bytes
r 2012 R2 Standard Evaluation
ows License valid for 180 days
Build 9600
FRA
US
10:53
14-10-21

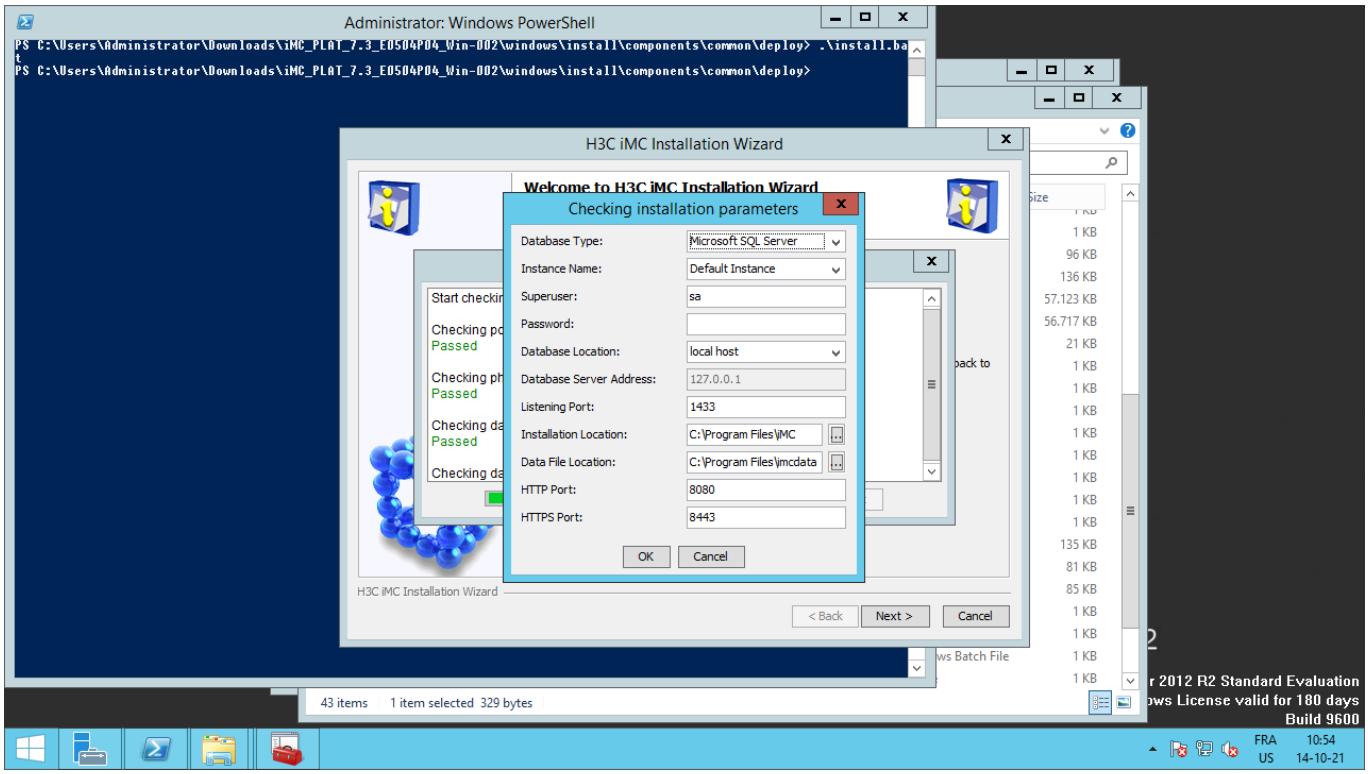

Administrator: Windows PowerShell
PS C:\Users\Administrator\Downloads\iMC_PLAT_7.3_E0504P04_Win-002\windows\install\components\common\deploy> .\install.bat
PS C:\Users\Administrator\Downloads\iMC_PLAT_7.3_E0504P04_Win-002\windows\install\components\common\deploy>
43 items
1 item selected 329 bytes
r 2012 R2 Standard Evaluation
ows License valid for 180 days
Build 9600
FRA
US
10:53
14-10-21


Administrator: Windows PowerShell
PS C:\Users\Administrator\Downloads\iMC_PLAT_7.3_E0504P04_Win-002\windows\install\components\common\deploy> .\install.bat
PS C:\Users\Administrator\Downloads\iMC_PLAT_7.3_E0504P04_Win-002\windows\install\components\common\deploy>
Select Country/Region, Language, and Installation Type
Country/Region: Belgique
Language: English
Installation Type: Typical - Deploys all platform components on the local server
Custom - Customizes components or deploys components in distributed mode
Wizard
You are installing the platform now. To install new components, use the following function in the Intelligent Deployment Monitoring Agent after you install the platform.
Intelligent Deployment Monitoring Agent
Monitor
Process
Deploy
Environment
Start iMC
Stop iMC
Install
Automatically start the services when the OS starts.
Exit
Disk Usage
Partition
Total Space(KB)
Used Space(KB)
Free Space(KB)
Usage
C:\
41,929,616
23,821,432
18,108,184
56.81%
OK
43 items
1 item selected 329 bytes
r 2012 R2 Standard Evaluation
ows License valid for 180 days
Build 9600
FRA
US
10:54
14-10-21

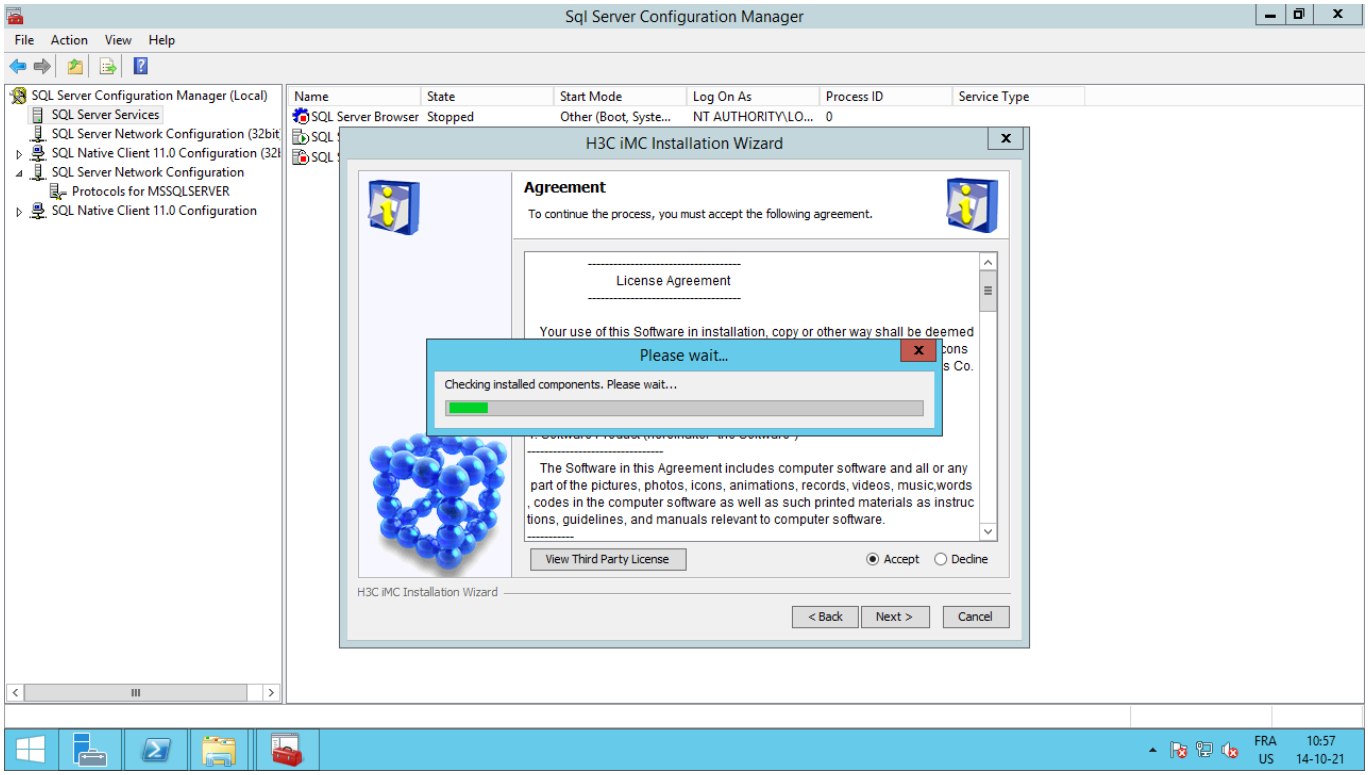

Administrator: Windows PowerShell
PS C:\Users\Administrator\Downloads\iMC_PLAT_7.3_E0504P04_Win-002\windows\install\components\common\deploy> .\install.bat
PS C:\Users\Administrator\Downloads\iMC_PLAT_7.3_E0504P04_Win-002\windows\install\components\common\deploy>
Choose folder
Components location
Browse...
Data backup by using DBMan is recommended before installation or upgrade.
OK
Cancel
Size
43 items
1 item selected 329 bytes
r 2012 R2 Standard Evaluation
ows License valid for 180 days
Build 9600
FRA
US
10:54
14-10-21

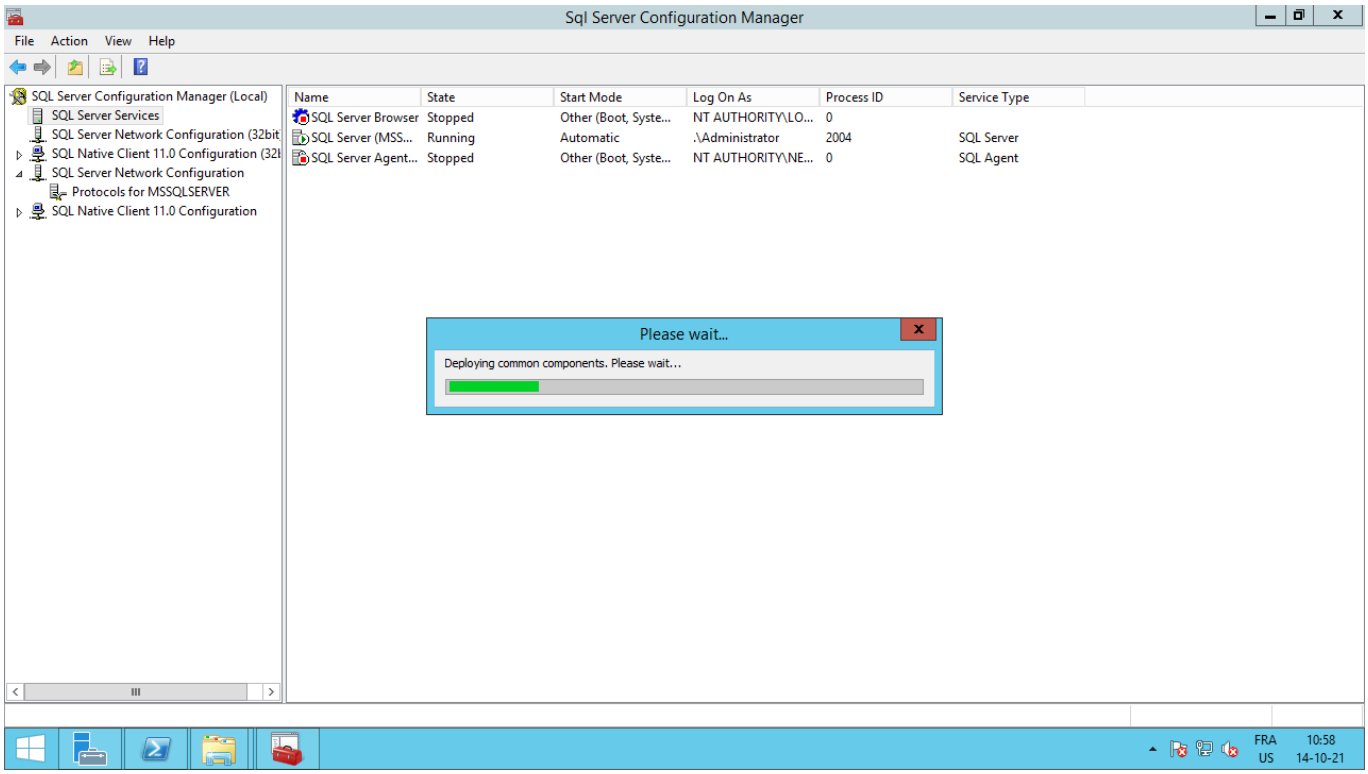

Administrator: Windows PowerShell
PS C:\Users\Administrator\Downloads\iMC_PLAT_7.3_E0504P04_Win-002\windows\install\components\common\deploy> .\install.bat
PS C:\Users\Administrator\Downloads\iMC_PLAT_7.3_E0504P04_Win-002\windows\install\components\common\deploy>
Choose folder
Rechercher dans :
iMC_PLAT_7.3_E0504P04_Win-002
manual
windows
Recent Items
Desktop
Documents
This PC
Network
Nom du dossier :
ministrator\Downloads\iMC_PLAT_7.3_E0504P04_Win-002
OK
Fichiers du type :
Tous les fichiers
Annuler
Size
43 items
1 item selected 329 bytes
r 2012 R2 Standard Evaluation
ows License valid for 180 days
Build 9600
FRA
US
10:54
14-10-21



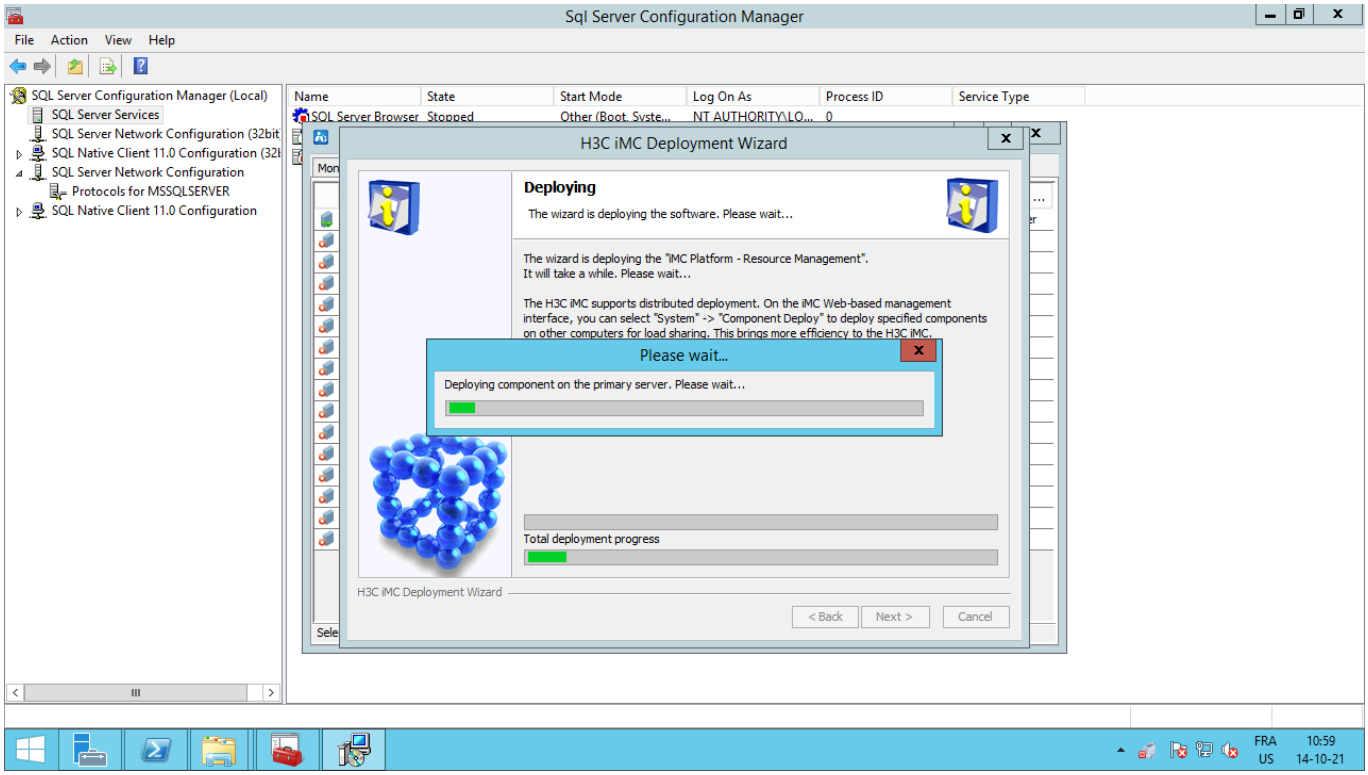


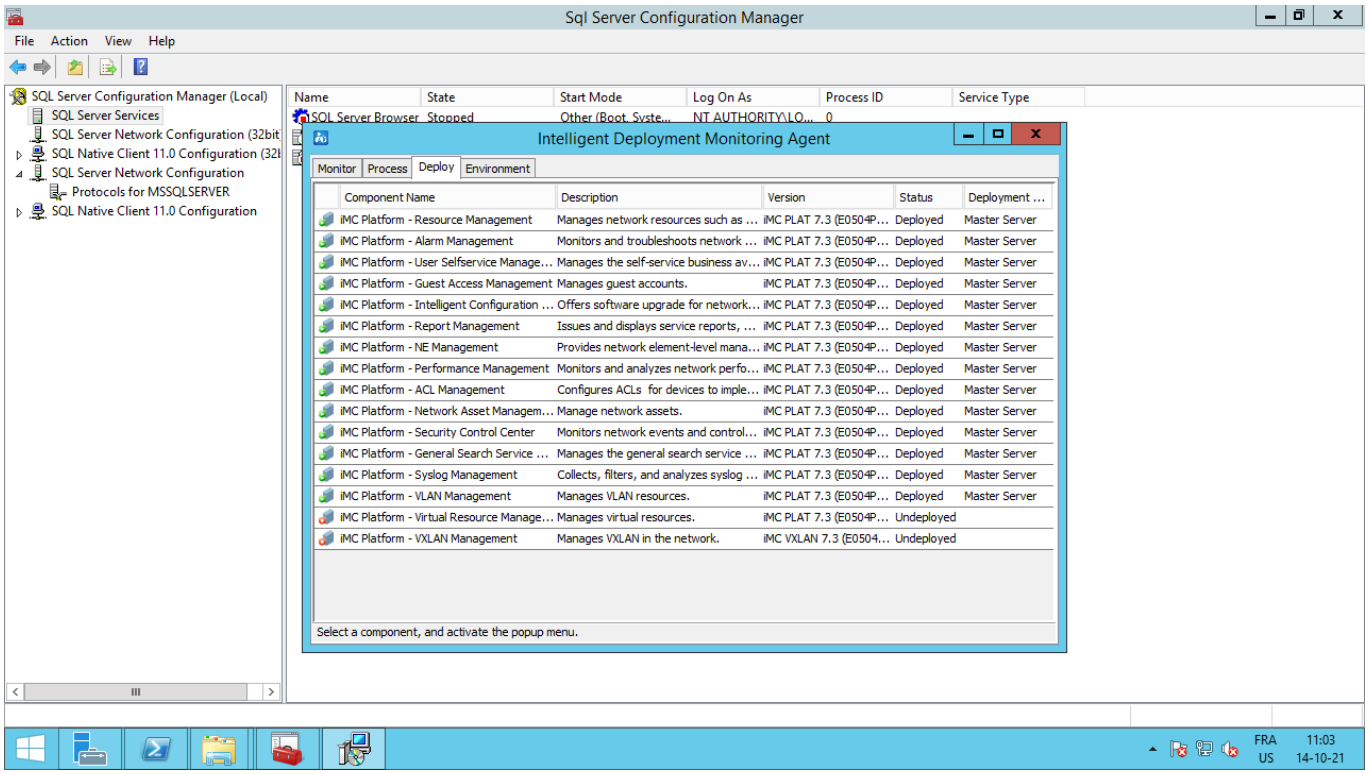
Fill in the password set up at the install of the SQLServer.

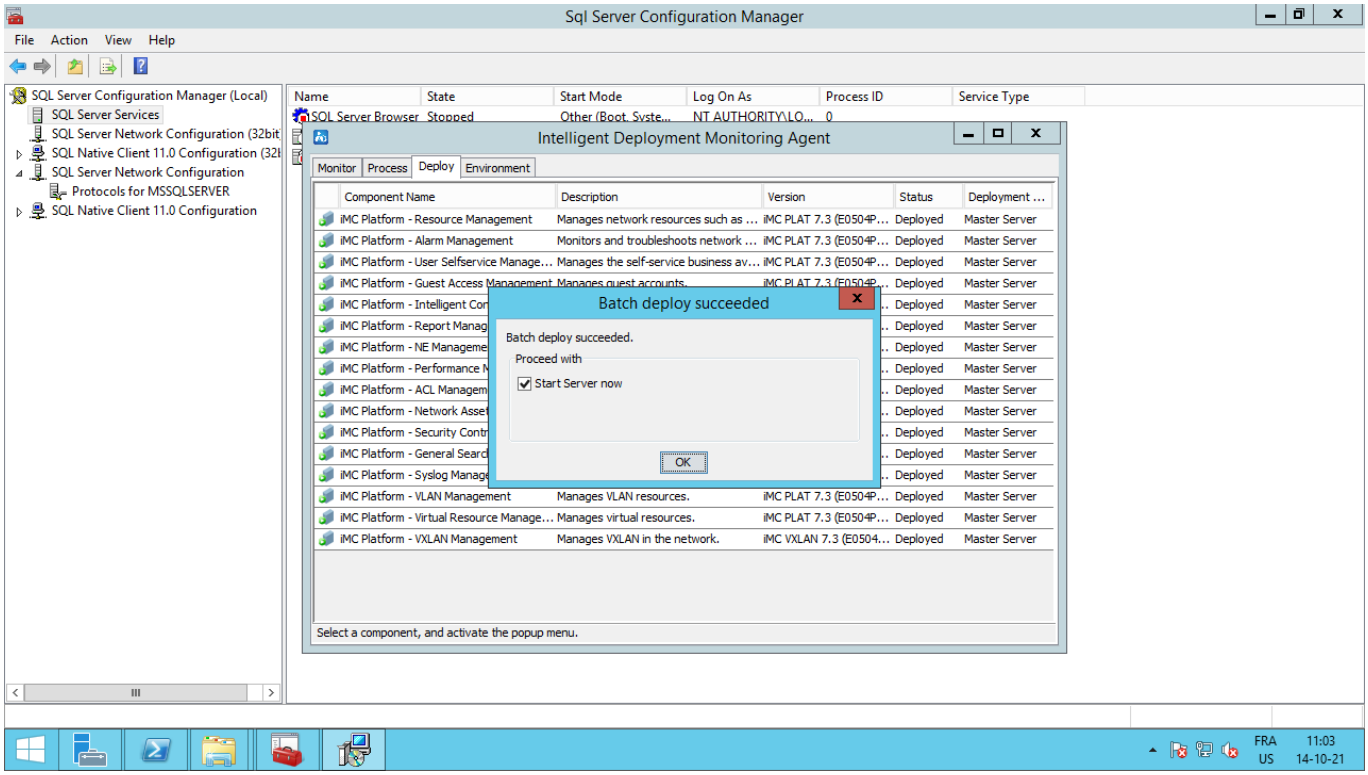


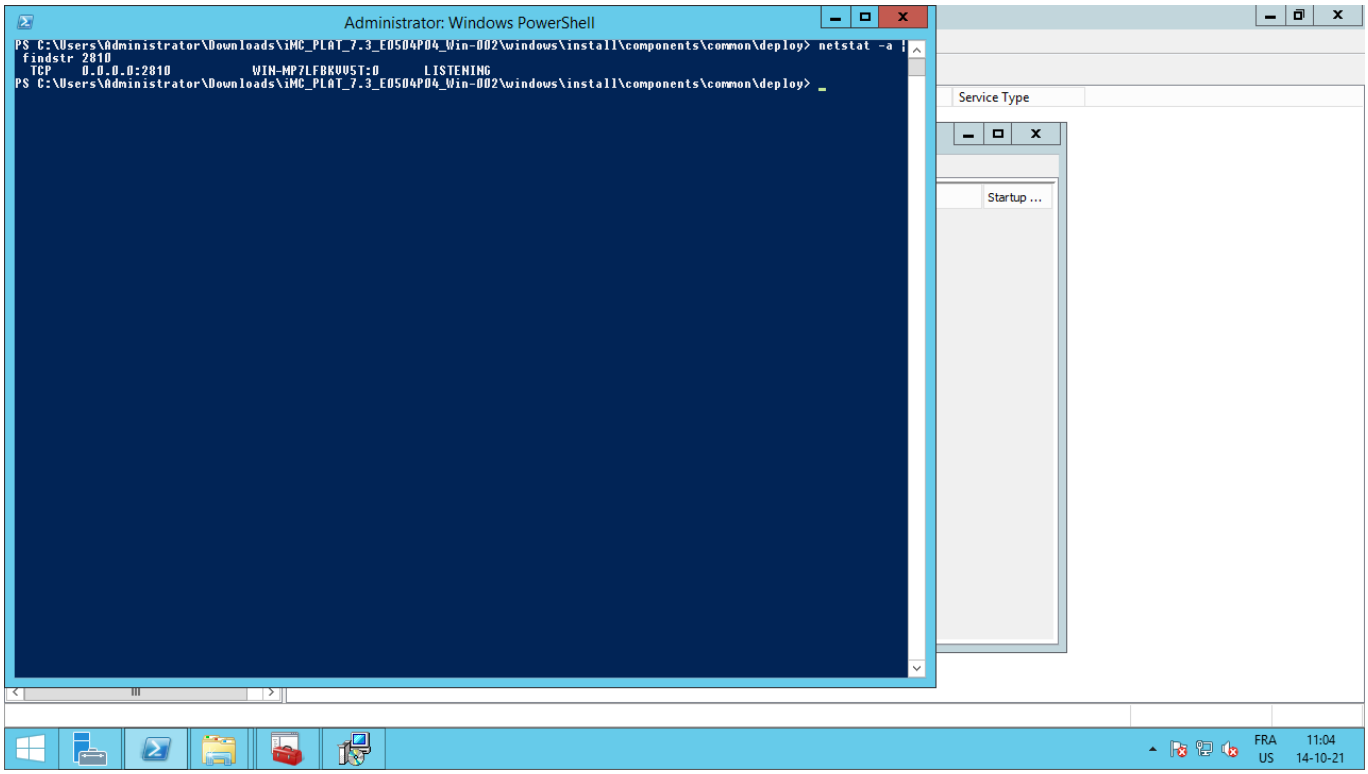


Note: the fact that the *SQL Server Agent* service doesn't run seems to be a normal behaviour because of the the db versions is an "express".


Sql Server Configuration Manager
File Action View Help
SQL Server Configuration Manager (Local)
SQL Server Services
SQL Server Network Configuration (32bit
SQL Native Client 11.0 Configuration (32
SQL Server Network Configuration
Protocols for MSSQLSERVER
SQL Native Client 11.0 Configuration
Name State Start Mode Log On As Process ID Service Type
SQL Server Browser Stopped Other (Boot, Syste... NT AUTHORITY\LO... 0
H3C iMC Deployment Wizard
Deploying
The wizard is deploying the software. Please wait...
The wizard is deploying the "iMC Platform - Resource Management".
It will take a while. Please wait...
The H3C iMC supports distributed deployment. On the iMC Web-based management interface, you can select "System" -> "Component Deploy" to deploy specified components on other computers for load sharing. This brings more efficiency to the H3C iMC.
Please wait...
Deploying component on the primary server. Please wait...
Total deployment progress
H3C iMC Deployment Wizard
< Back
Next >
Cancel
FRA
US
10:59
14-10-21


Sql Server Configuration Manager
File Action View Help
SQL Server Configuration Manager (Local)
SQL Server Services
SQL Server Network Configuration (32bit
SQL Native Client 11.0 Configuration (32
SQL Server Network Configuration
Protocols for MSSQLSERVER
SQL Native Client 11.0 Configuration
Name State Start Mode Log On As Process ID Service Type
SQL Server Browser Stopped Other (Boot, Syste... NT AUTHORITY\LO... 0
Intelligent Deployment Monitoring Agent
Monitor Process Deploy Environment
Component Name Description Version Status Deployment ...
iMC Platform - Resource Management Manages network resources such as ... iMC PLAT 7.3 (E0504P... Deployed Master Server
iMC Platform - Alarm Management Monitors and troubleshoots network ... iMC PLAT 7.3 (E0504P... Deployed Master Server
iMC Platform - User Selfservice Manage... Manages the self-service business av... iMC PLAT 7.3 (E0504P... Deployed Master Server
iMC Platform - Guest Access Management Manages guest accounts. iMC PLAT 7.3 (E0504P... Deployed Master Server
iMC Platform - Intelligent Configuration ... Offers software upgrade for network... iMC PLAT 7.3 (E0504P... Deployed Master Server
iMC Platform - Report Management Issues and displays service reports, ... iMC PLAT 7.3 (E0504P... Deployed Master Server
iMC Platform - NE Management Provides network element-level mana... iMC PLAT 7.3 (E0504P... Deployed Master Server
iMC Platform - Performance Management Monitors and analyzes network perfo... iMC PLAT 7.3 (E0504P... Deployed Master Server
iMC Platform - ACL Management Configures ACLs for devices to imple... iMC PLAT 7.3 (E0504P... Deployed Master Server
iMC Platform - Network Asset Managem... Manage network assets. iMC PLAT 7.3 (E0504P... Deployed Master Server
iMC Platform - Security Control Center Monitors network events and control... iMC PLAT 7.3 (E0504P... Deployed Master Server
iMC Platform - General Search Service ... Manages the general search service ... iMC PLAT 7.3 (E0504P... Deployed Master Server
iMC Platform - Syslog Management Collects, filters, and analyzes syslog ... iMC PLAT 7.3 (E0504P... Deployed Master Server
iMC Platform - VLAN Management Manages VLAN resources. iMC PLAT 7.3 (E0504P... Deployed Master Server
iMC Platform - Virtual Resource Manage... Manages virtual resources. iMC PLAT 7.3 (E0504P... Undeployed
iMC Platform - VXLAN Management Manages VXLAN in the network. iMC VXLAN 7.3 (E0504... Undeployed
Select a component, and activate the popup menu.
FRA
US
11:03
14-10-21





We can see *"dbman"*.

Analysis Setup

In order to analyze ZDI-17-836 / CVE-12561 :

https://www.zerodayinitiative.com/advisories/ZDI-17-836/

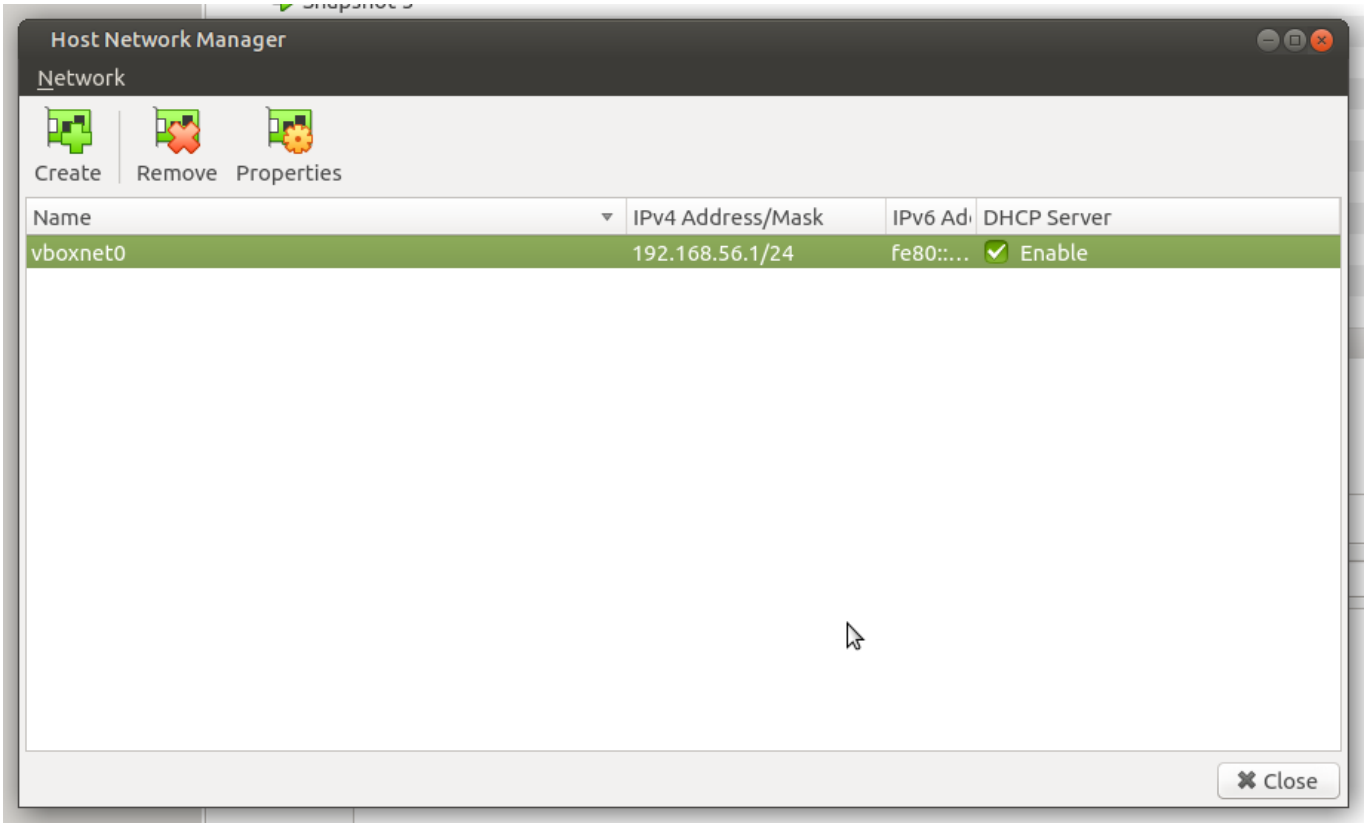
the use of the following software/tools is of great assistance.

Preliminary Virtual Network Configuration

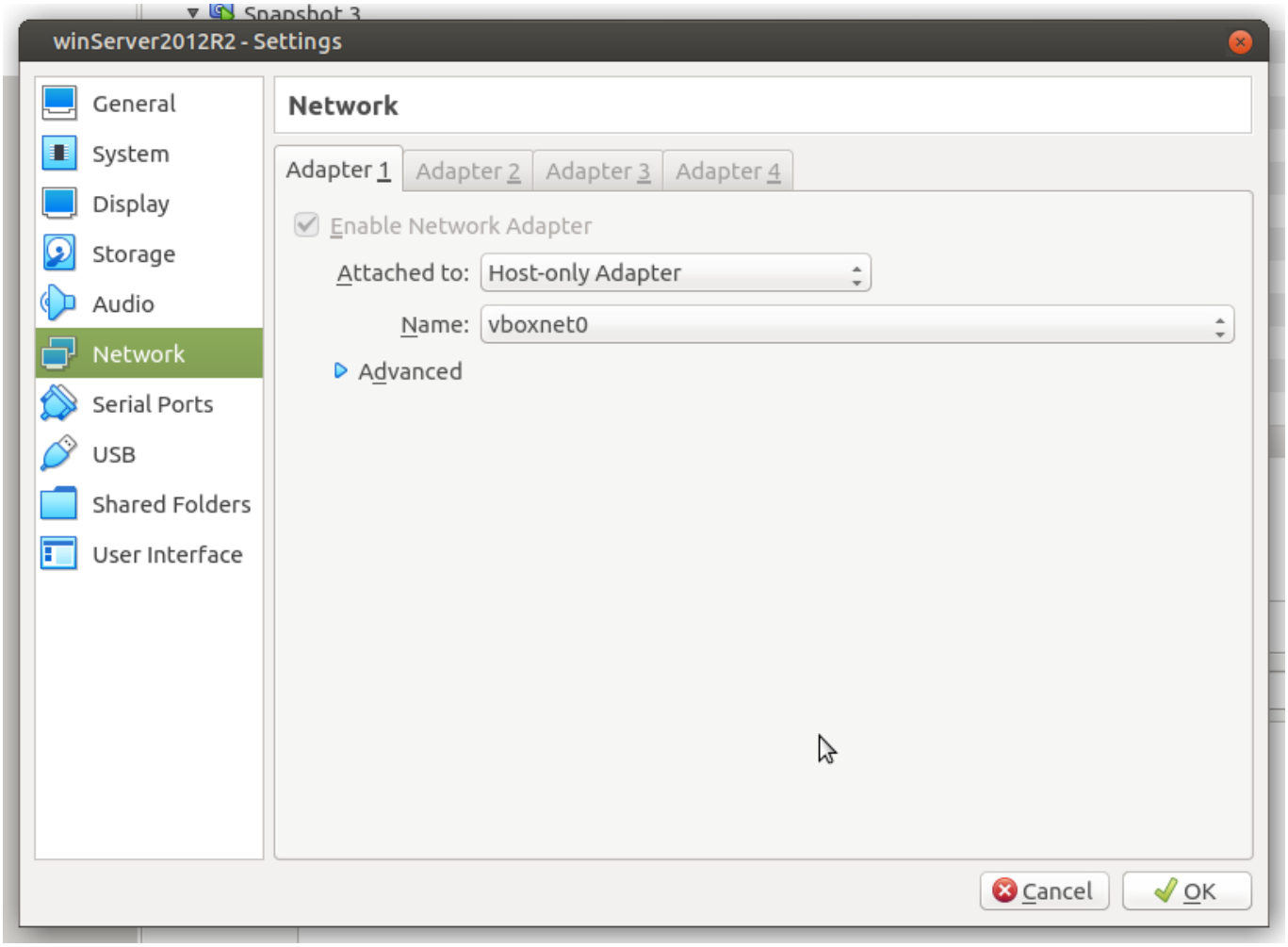
I am going with the following network setup:

To somehow isolate the network of the WinServer2012R2 and potential other instances, I am adding and using "Host-only adapter" for the VMs.

In Oracle VM VirtualBox Manager, under *File*, *Host Network Manager...*, create an adapter:



Using the created adapter in the concerned VMs:



WindowsServer2012R2

On the iMC hosting machine(WindowsServer2012R2):

I am using the following software/setup:

- WinDbg 10.0.22000.194 + msecdbg
- Ghidra-SRE ghidra_10.0.4_PUBLIC_20210928.zip
- xdbg snapshot_2021-07-01_23-17.zip
- BinDiff7 https://zynamics.com/software.html
- ProcessHacker https://processhacker.sourceforge.io/
- Wireshark(win64-3.4.9) https://www.wireshark.org/
- ...

WinDbg Install

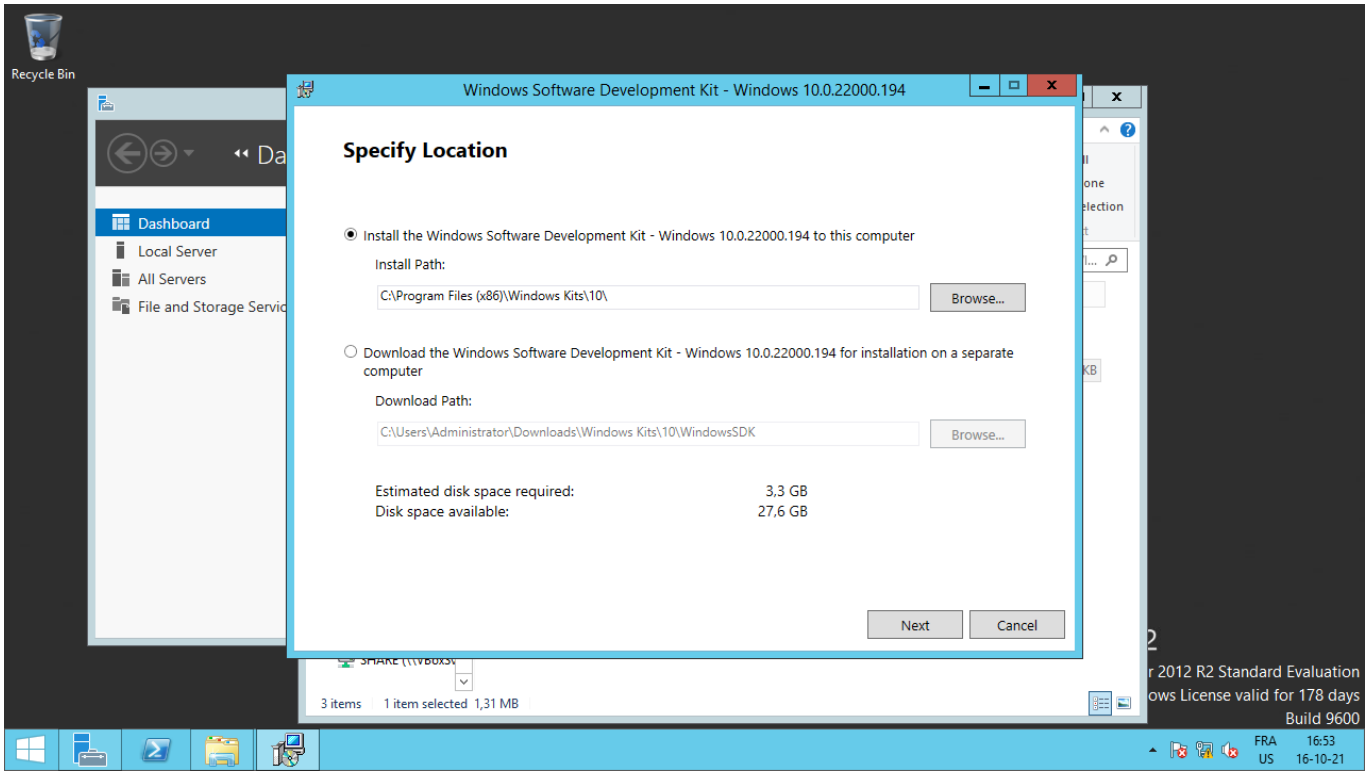
I am going for the iso version installer:

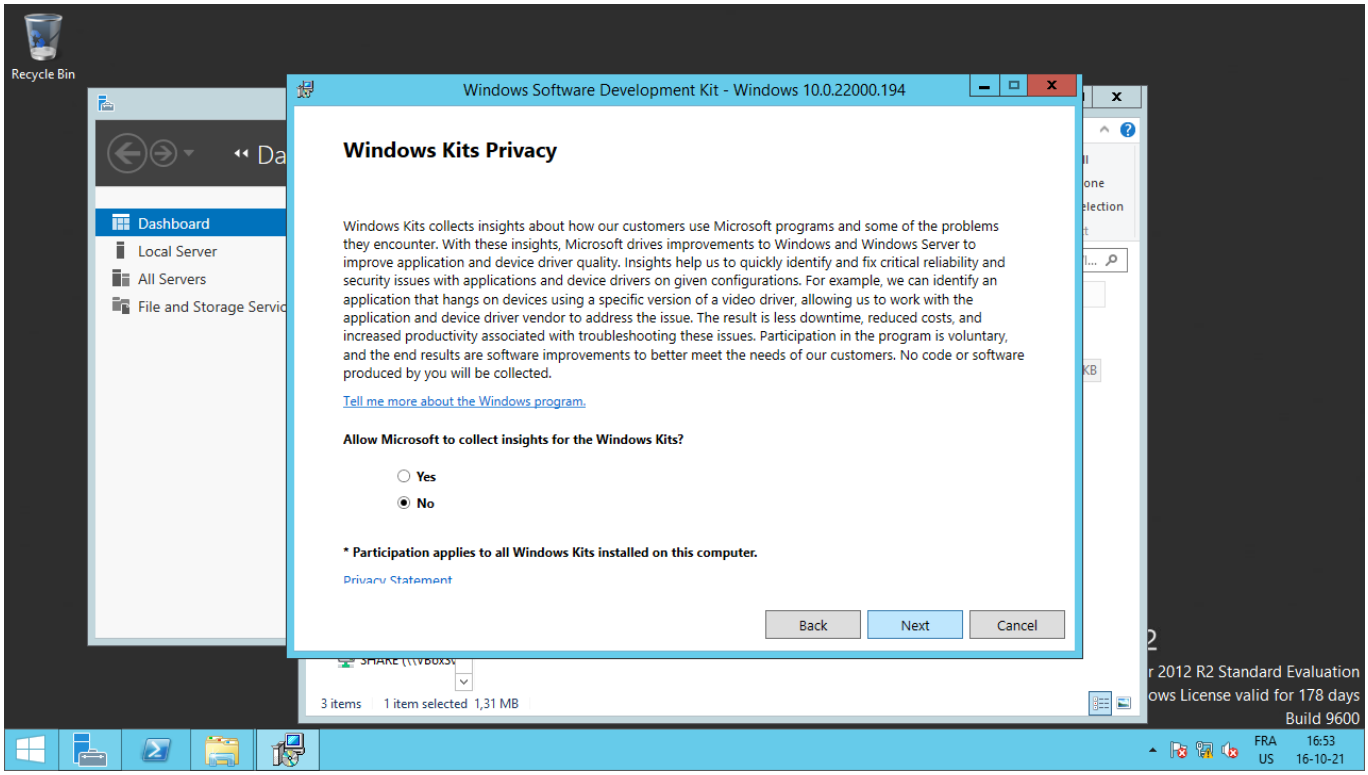
https://developer.microsoft.com/en-us/windows/downloads/windows-sdk/

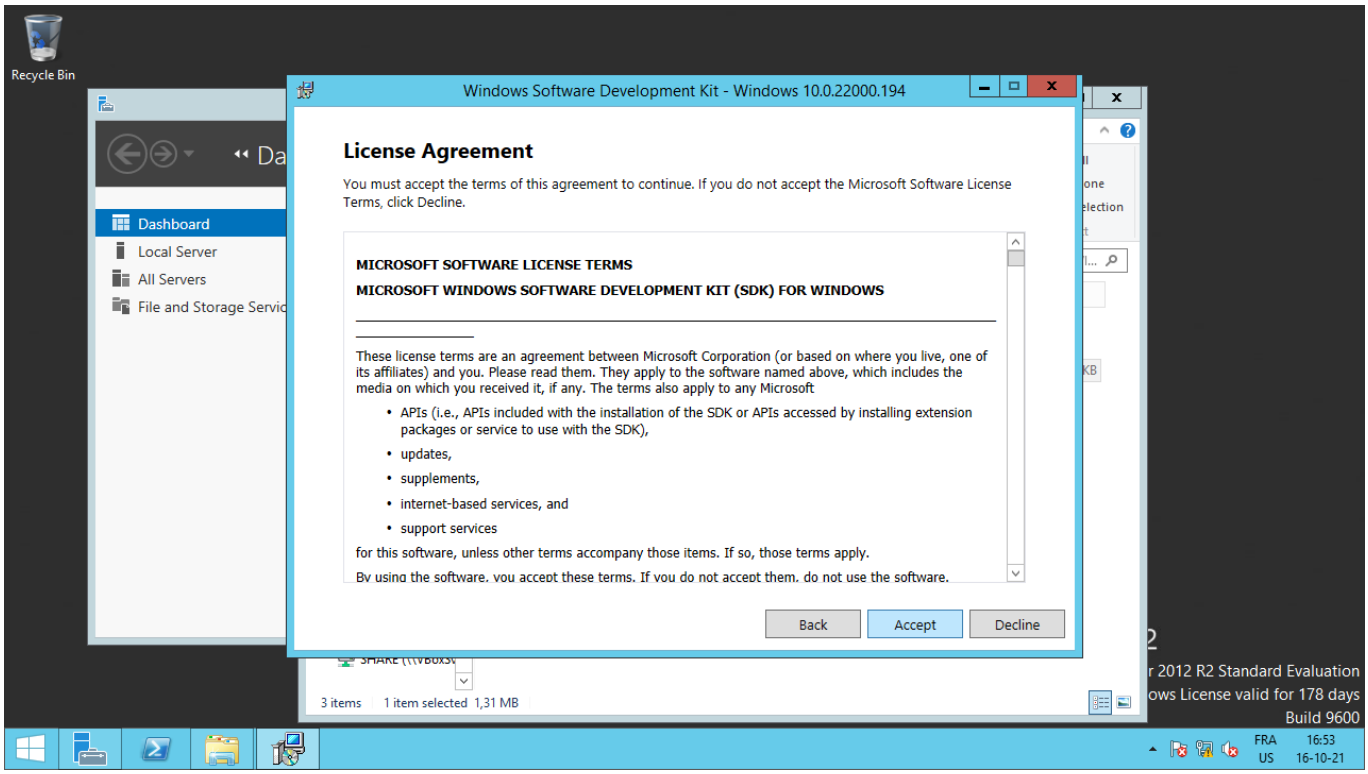
```
$ sha1sum 22000.194.210911-1543.co_release_svc_prod1_WindowsSDK.iso
b875d4a935b08493a0e35f523010854abf2884c3  22000.194.210911-
1543.co_release_svc_prod1_WindowsSDK.iso
```

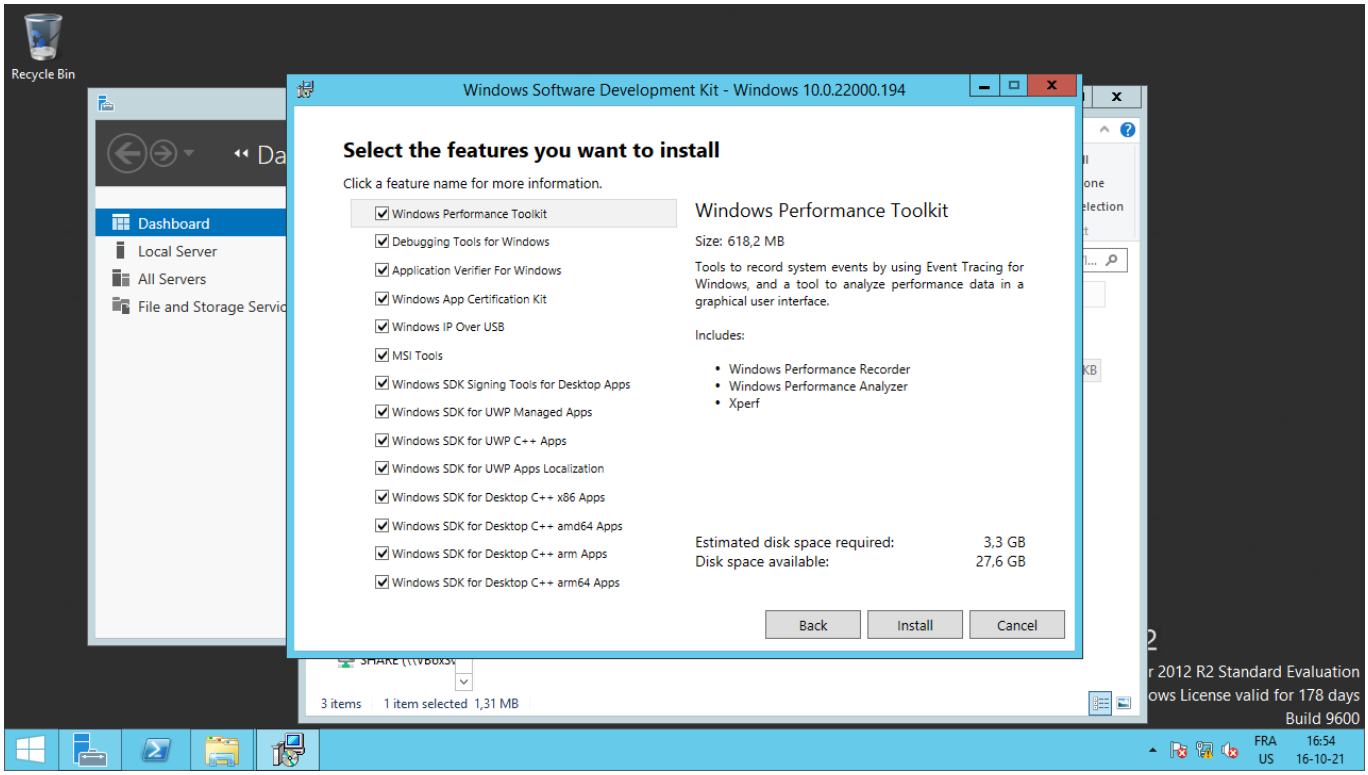
To install through the iso:

- right click on the iso, then select *Mount*
- access the disk through the Windows Explorer and launch the installation




Recycle Bin
Windows Software Development Kit - Windows 10.0.22000.194
Windows Kits Privacy
Windows Kits collects insights about how our customers use Microsoft programs and some of the problems they encounter. With these insights, Microsoft drives improvements to Windows and Windows Server to improve application and device driver quality. Insights help us to quickly identify and fix critical reliability and security issues with applications and device drivers on given configurations. For example, we can identify an application that hangs on devices using a specific version of a video driver, allowing us to work with the application and device driver vendor to address the issue. The result is less downtime, reduced costs, and increased productivity associated with troubleshooting these issues. Participation in the program is voluntary, and the end results are software improvements to better meet the needs of our customers. No code or software produced by you will be collected.
Tell me more about the Windows program.
Allow Microsoft to collect insights for the Windows Kits?
Yes
No
* Participation applies to all Windows Kits installed on this computer.
Privacy Statement
Back
Next
Cancel
Dashboard
Local Server
All Servers
File and Storage Servic
3 items
1 item selected 1,31 MB
r 2012 R2 Standard Evaluation
ows License valid for 178 days
Build 9600
FRA
US
16:53
16-10-21


Recycle Bin
Windows Software Development Kit - Windows 10.0.22000.194
License Agreement
You must accept the terms of this agreement to continue. If you do not accept the Microsoft Software License Terms, click Decline.
MICROSOFT SOFTWARE LICENSE TERMS
MICROSOFT WINDOWS SOFTWARE DEVELOPMENT KIT (SDK) FOR WINDOWS
These license terms are an agreement between Microsoft Corporation (or based on where you live, one of its affiliates) and you. Please read them. They apply to the software named above, which includes the media on which you received it, if any. The terms also apply to any Microsoft
APIs (i.e., APIs included with the installation of the SDK or APIs accessed by installing extension packages or service to use with the SDK),
updates,
supplements,
internet-based services, and
support services
for this software, unless other terms accompany those items. If so, those terms apply.
By using the software, you accept these terms. If you do not accept them, do not use the software.
Back
Accept
Decline
Dashboard
Local Server
All Servers
File and Storage Servic
3 items
1 item selected 1,31 MB
r 2012 R2 Standard Evaluation
ows License valid for 178 days
Build 9600
FRA
US
16:53
16-10-21



Msecdbg extensions

The **msecdbg** winDbg extensions helps one in assessing if an observed behaviour in the debugger(generally a crash) is exploitable or not.

The following is the official link:

https://archive.codeplex.com/?p=msecdbg

I am going for a compiled version, here:

https://blog.didierstevens.com/2018/07/17/exploitable-crash-analyzer-statically-linked-crt/
https://didierstevens.com/files/software/MSECWinDbgExtensions.zip

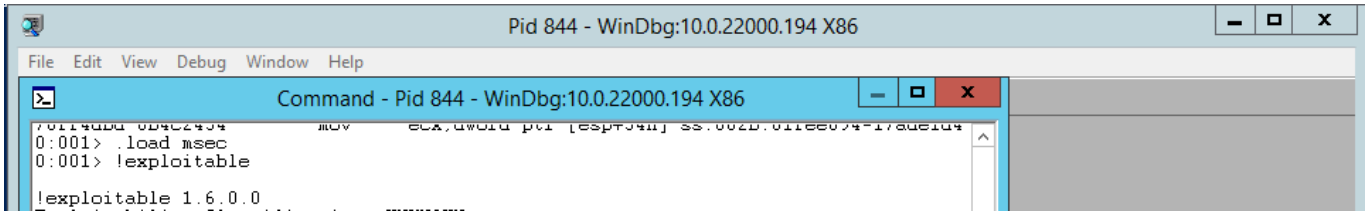
```
$ sha1sum MSECWinDbgExtensions.zip
63741f82e4da2b6762776b7bcf717c4497c05cca  MSECWinDbgExtensions.zip
```

Installation:

The installation consists in placing the *MSEC.dll*(selecting the one corresponding to your ISA) in the following directory:

```
C:\Program Files (x86)\Windows Kits\10\Debuggers\x86\winext
```

The msecdbg extension can be loaded as follow:



Wireshark

The installation is pretty much straightforward, just need to go for the default configuration.

https://www.wireshark.org/download.html

```
$ sha1sum Wireshark-win64-3.4.9.exe
fea985c482130a2f92b04e2e3f8bbeb8815c3f5f  Wireshark-win64-3.4.9.exe
```

Kali 2021.3

I am using a Kali 2021.3 VM for a quick access to additional tools.
Kali version:

```
┌──(kali㉿kali)-[~]
└─$ lsb_release -a
No LSB modules are available.
Distributor ID: Kali
Description:    Kali GNU/Linux Rolling
Release:    2021.3
Codename:   kali-rolling
```

Ghidra-SRE

Ghidra 10.0.4:

```
$ sha256sum ghidra_10.0.4_PUBLIC_20210928.zip
1ce9bdf2d7f6bdfe5dccd06da828af31bc74acfd800f71ade021d5211e820d5e
ghidra_10.0.4_PUBLIC_20210928.zip
```

Written mostly in Java, the install comes down to decompressing the archive and make the JDK visible to it.
Installation Guide

BinDiff 7

```
$ sha1sum bindiff7.msi
95550a1539eabc8f91066210c815ac756eaa288f  bindiff7.msi
```

https://zynamics.com/software.html

For nice graph visualization, using BinExport for Ghidra plugin, we can export the files(.BinExport) that can be fed into BinDiff.

https://github.com/google/binexport

Main reference for this setup:
https://ihack4falafel.github.io/Patch-Diffing-with-Ghidra/

Root Cause Analysis(RCA)

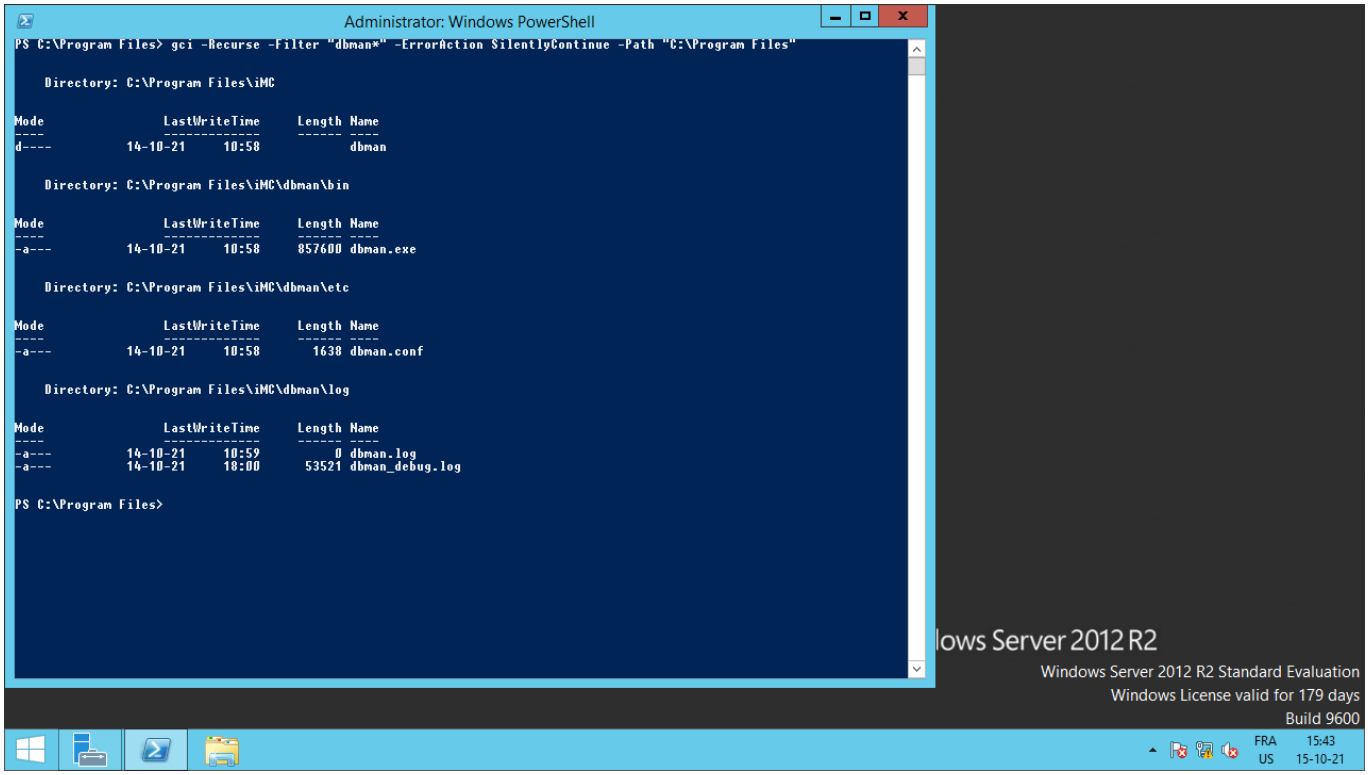
Dbman.exe

From https://www.zerodayinitiative.com/advisories/ZDI-17-836/ :

```
This vulnerability allows remote attackers to execute arbitrary code on
vulnerable installations of Hewlett Packard Enterprise Intelligent Management
Center. Authentication is not required to exploit this vulnerability.

The specific flaw exists within dbman service, which listens on TCP port 2810 by
default. A crafted opcode 10012 message can cause a pointer to be reused after
it has been freed. An attacker can leverage this vulnerability to execute code
under the context of SYSTEM.
```

We are interested in *dbman.exe*, which is a utility that deals with database related operations.

After the iMC's installation *dbman.exe* is located at:



In this setting. *dbman.exe* binary lies in:

```
C:\Program Files\iMC\dbman\bin
```

dbman.exe versions hashes:

```
ixidor@pr0c:~/SHARE/dbman-iMC_PLAT_7.3_E0504_Ent_Win-003/dbman/bin$ md5sum dbman.exe
c0a5cd15339a8eda718886510a347ce8  dbman.exe
ixidor@pr0c:~/SHARE/dbman-iMC_PLAT_7.3_E0504_Ent_Win-003/dbman/bin$ cd ../../../dbman-iMC_PLAT_7.3_E0504P04_Win-002/
ixidor@pr0c:~/SHARE/dbman-iMC_PLAT_7.3_E0504P04_Win-002$ md5sum dbman/bin/dbman.exe
3cd632d4245d08e76014e7240e5f5f82  dbman/bin/dbman.exe
ixidor@pr0c:~/SHARE/dbman-iMC_PLAT_7.3_E0504P04_Win-002$
```

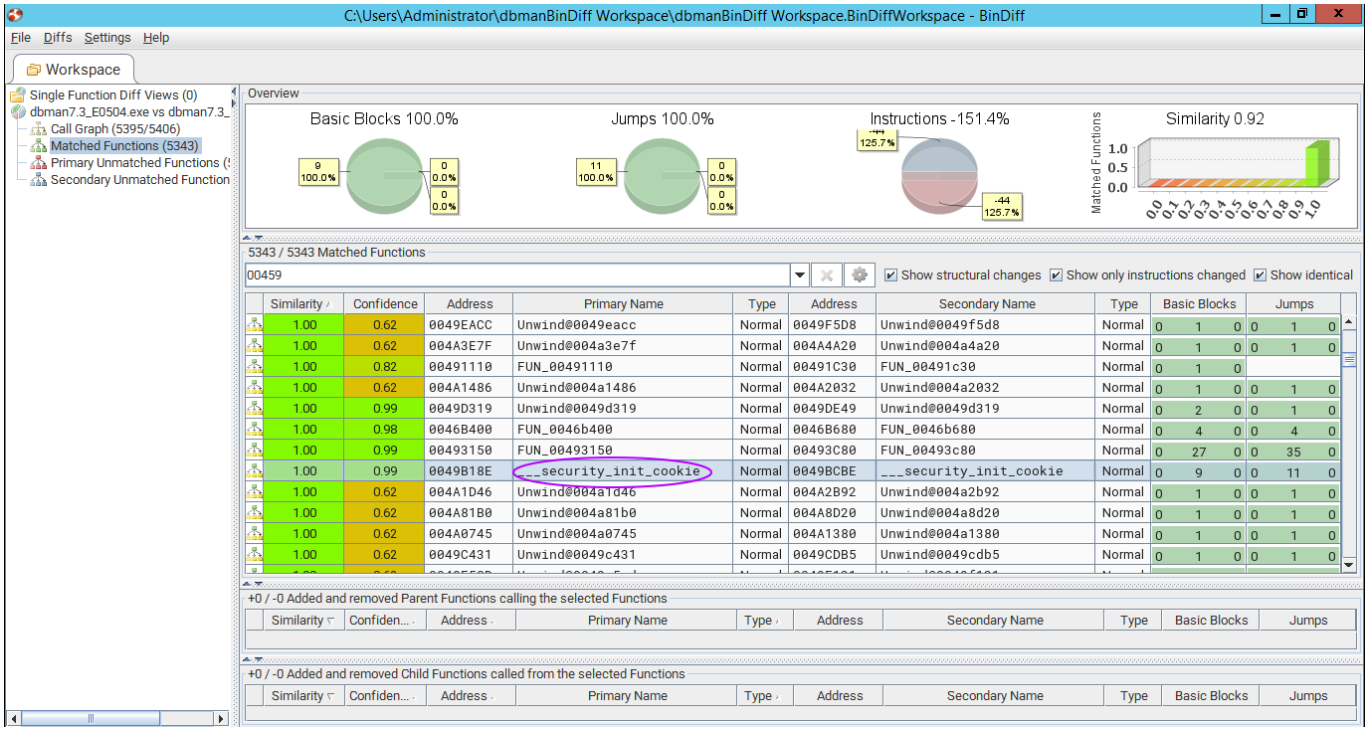
PE File Security Mitigations:

It's always interesting to have a first idea of the PE file(*dbman.exe*) security mitigations that are applied or not:

```
ixidor@pr0c:~/SHARE/build$ ./winchecksec ../dbman-iMC_PLAT_7.3_E0504_Ent_Win-003/dbman/bin/dbman.exe
Results for: ../dbman-iMC_PLAT_7.3_E0504_Ent_Win-003/dbman/bin/dbman.exe
Dynamic Base     : "NotPresent"
ASLR             : "NotPresent"
High Entropy VA  : "NotPresent"
Force Integrity  : "NotPresent"
Isolation        : "Present"
NX               : "NotPresent"
SEH              : "Present"
CFG              : "NotPresent"
RFG              : "NotPresent"
SafeSEH          : "NotPresent"
GS               : "Present"
Authenticode     : "NotPresent"
.NET             : "NotPresent"

ixidor@pr0c:~/SHARE/build$ ./winchecksec ../dbman-iMC_PLAT_7.3_E0504P04_Win-002/dbman/bin/dbman.exe
Results for: ../dbman-iMC_PLAT_7.3_E0504P04_Win-002/dbman/bin/dbman.exe
Dynamic Base     : "NotPresent"
ASLR             : "NotPresent"
High Entropy VA  : "NotPresent"
Force Integrity  : "NotPresent"
Isolation        : "Present"
NX               : "NotPresent"
SEH              : "Present"
CFG              : "NotPresent"
RFG              : "NotPresent"
SafeSEH          : "NotPresent"
GS               : "Present"
Authenticode     : "NotPresent"
.NET             : "NotPresent"
```

At this point, I cannot say if the output of *winchecksec* is quite reliable and/or accurate.

It seems that the binary has been compiled with the /GS flag:



https://docs.microsoft.com/en-us/microsoft-365/security/defender-endpoint/exploit-protection-reference

Confirming the bug(s)

(ZDI-17-836 / CVE-12561)

Approaches

In the context where we have two(or more) pieces of software, where one version contains a weakness/vulnerability and the other some sort of a patch to mitigate this flaw. Then, probably one of the most straightforward approach to id it is a process called **binary diffing**. We're looking for similarities/discrepancies in the binaries.
More often in large code bases, bringing a patch can also introduce new bugs.

Another possible approach could be to use **taint analysis** to be able to pinpoint ; or at least, to narrow down the space search related to the weakness. In this particular case, it would be interesting to see if such an approach could yield the function of interest.
Given the architecture of iMC(Java and C++ code), I don't know if this approach could actually work.

There is the whole space of **dynamic binary instrumentation**(DBI), symbolic/concolic execution which has commonalities with the previous point.

Finally, whatever the approach chosen one can go with a blackbox or whitebox/greybox. In the context/mindset of exploit development, one is often interested in subverting/controlling designed functionalities. RE is a powerful but time consuming approach. The "exploitation" approach is way much more fun because what you're actually trying to do is to minimize the effort for a maximum gain(in theory) ; which in a way makes you some sort of an "optimized" programmer.

[ZDI-17-836 / CVE-12561]

To confirm the vulnerability or at least a bug, I am using the following python script(*fuzz.py*) from:
https://www.github.com/pwnslinger/exploit-repo

Often, it is easier to work backwards.

Given the information provided in the previous link, we should be able to identify two vulnerabilities:

- UAF(Use-After-Free)
- Stack-based buffer overflow

In both settings, I am using one Kali VM as the tester and a WindowsServer2012R2 VM(iMC software) as a testee.

UAF

To id UAF between the two versions of *dbman.exe*, we are going with a patch diffing process using Ghidra's *Version Tracking* to reduce and identify the number of potential function candidates ; using Bindiff to get a better visualization.

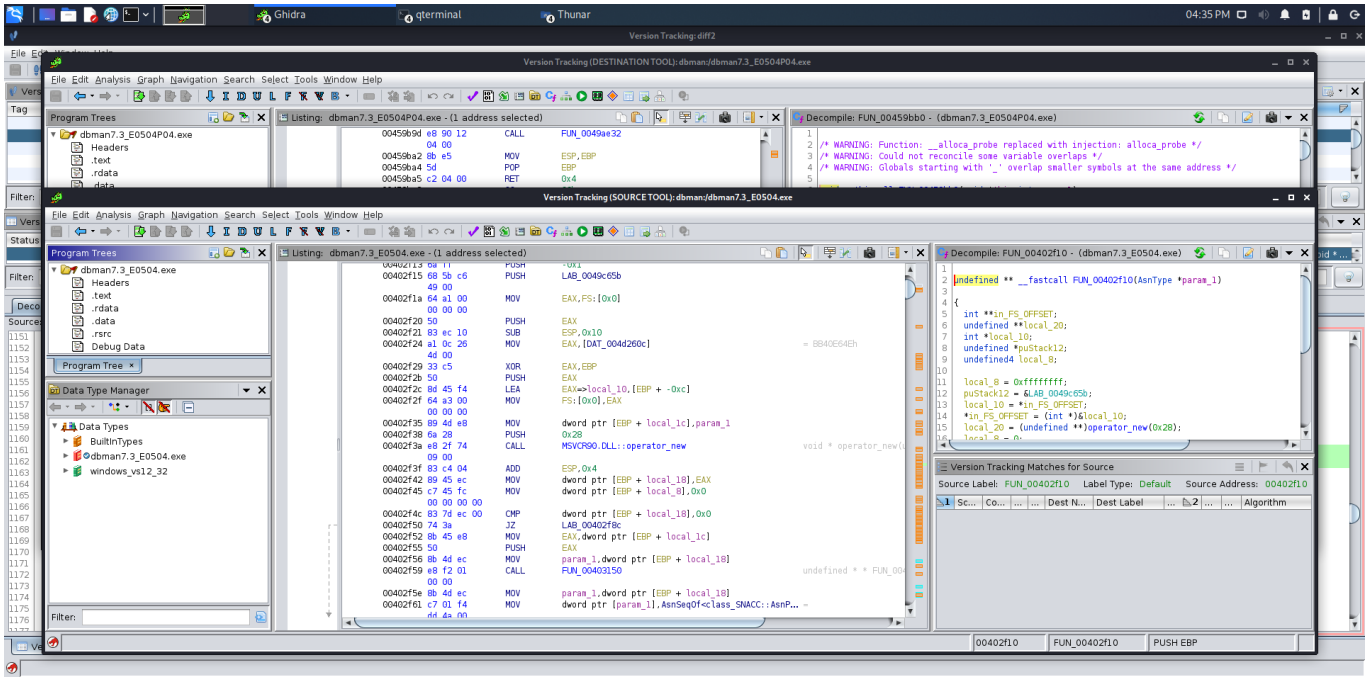
Given the nature of this class of bug, we are looking for something that roughly adds some check at some point ; something that frees and zeroes out a pointer or not, it depends from which perspective we're looking.

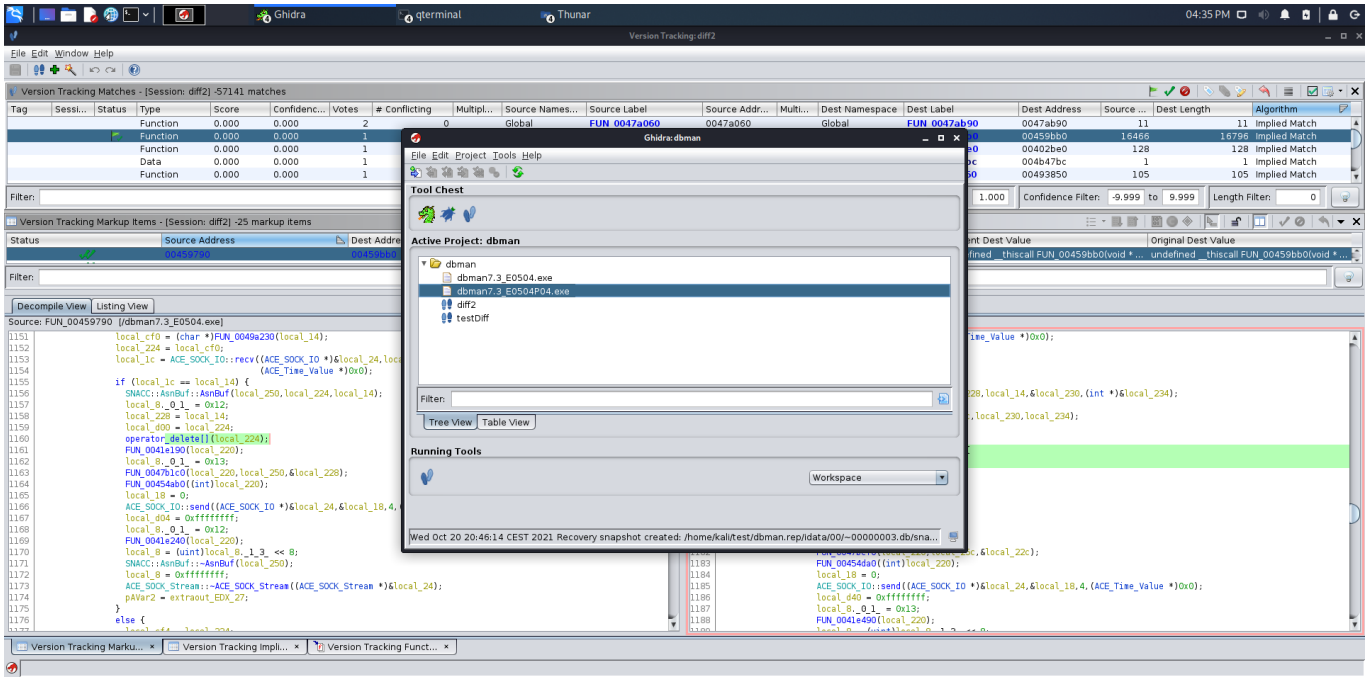
In this particular case, spotting the bug from a static only perspective isn't that easy because there is a call that is resolved dynamically i.e., when the binary is running. Combining static with a dynamic analysis/debugger helps.

Software:

Each *dbman.exe* from the installation of:
iMC_PLAT_7.3_E0504_Ent_Win-003 and iMC_PLAT_7.3_E0504P04_Win-002

Ingesting binaries into Ghidra before using Version Tracking:





A first attempt was to look for references of the opcode command:

```
>>> hex(10012)
'0x271c'
```

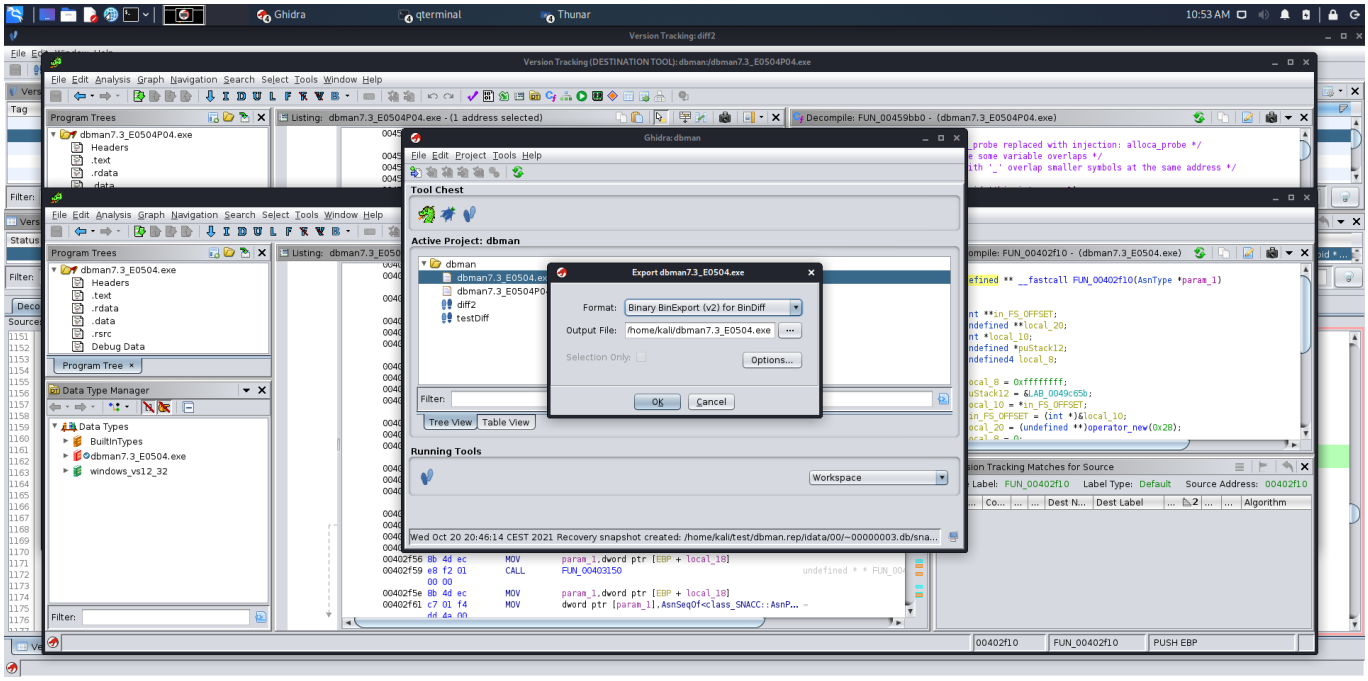
Didn't manage to get something useful by searching for this constant.

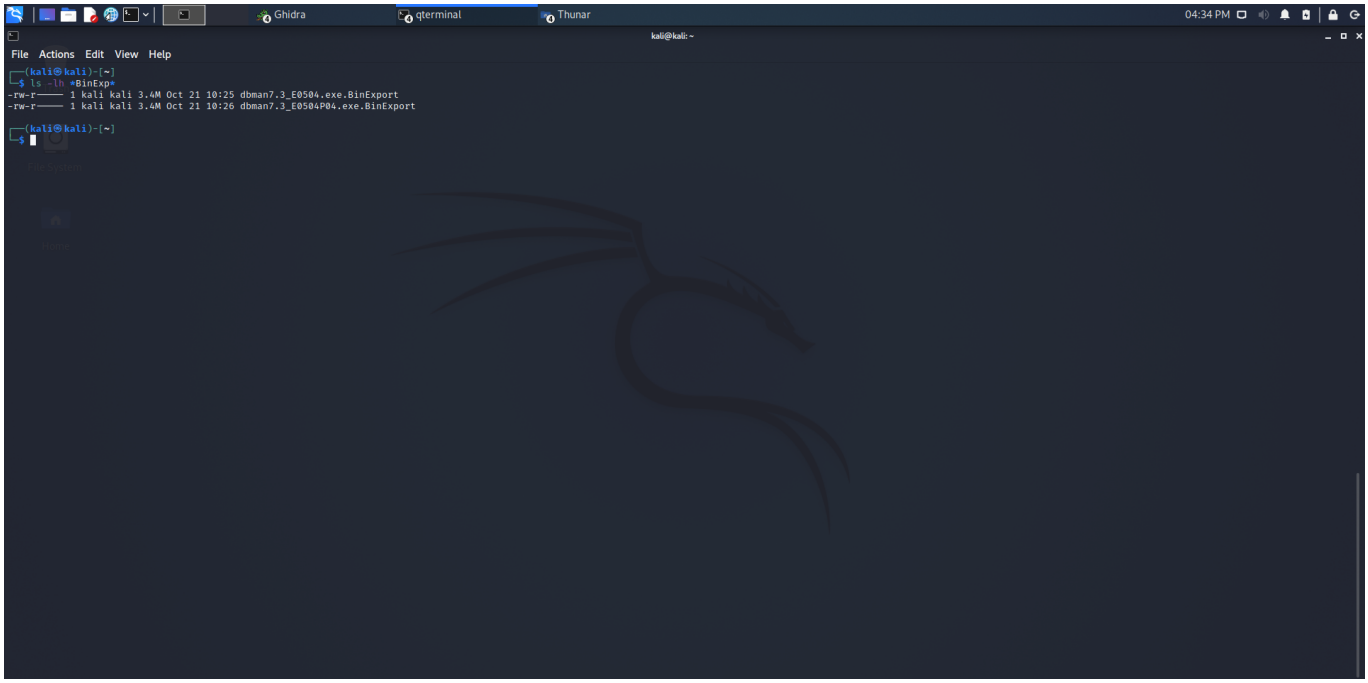
Speaking of opcode, it is likely that there is a typo in the *fuzz.py* script:

```
32 class AsnRemoteReservedFileRemove(object):
33     def __init__(self, reservedFilePath, backupPath, backFileExt, time):
34         super(AsnRemoteReservedFileRemove, self).__init__()
35         self.reservedFilePath = reservedFilePath
36         self.backupPath = backupPath
37         self.backFileExt = backFileExt
38         self.time = time
39         self.opode = 10012
40
41     def encode(self):
```

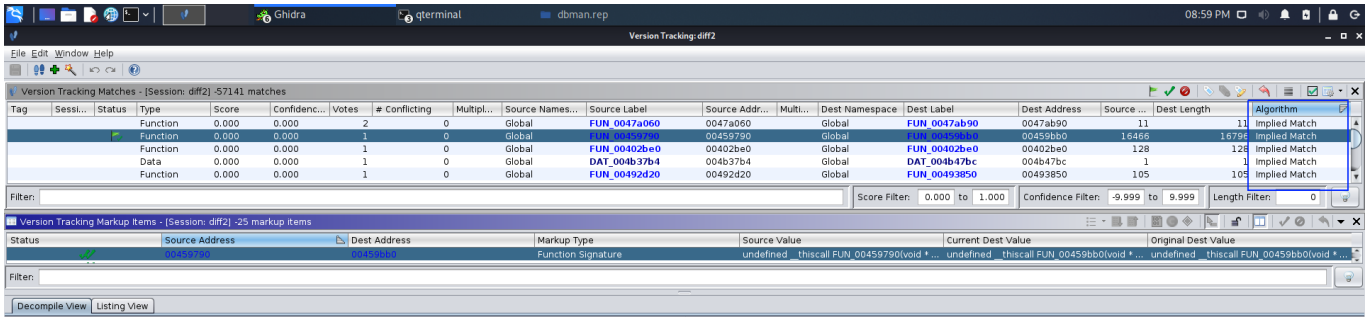
This should most likely be `self.opcode`.

Ghidra - BinExport for BinDiff visualization:



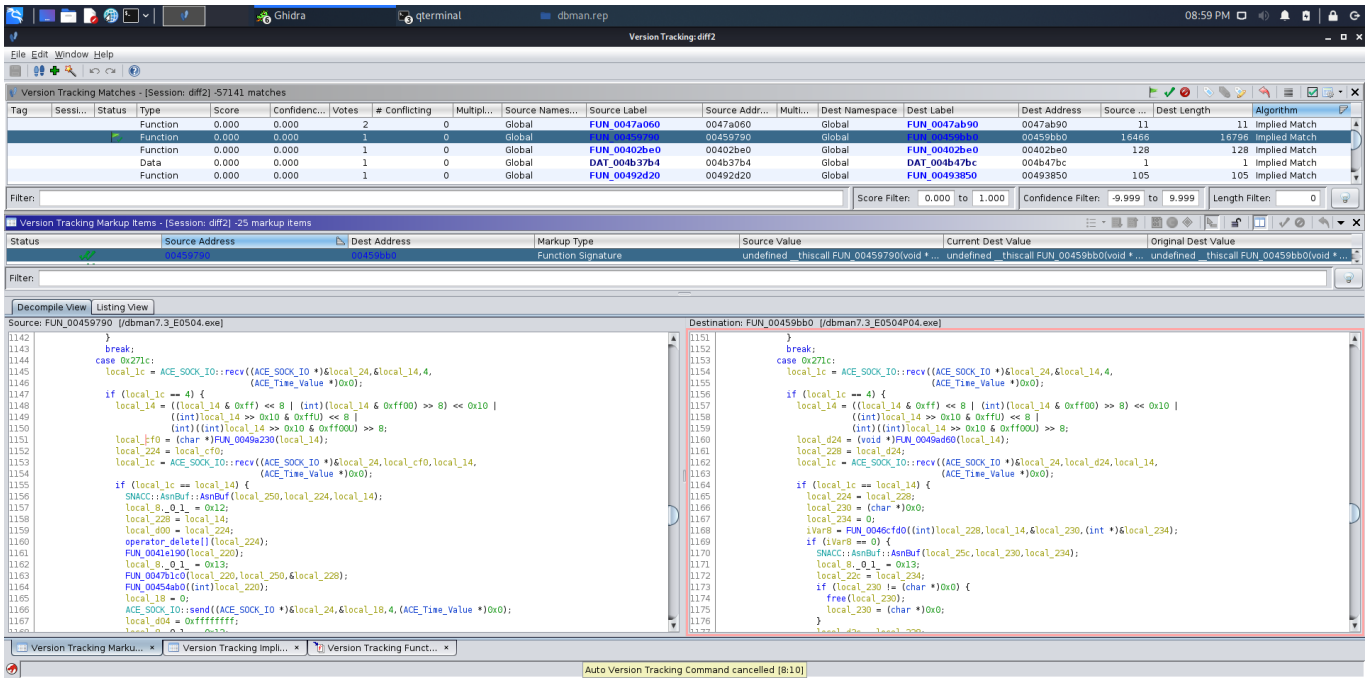


Back to Version Tracking, using and sorting by the *Implied Match* algorithm(maybe some sort of fuzzy matching):

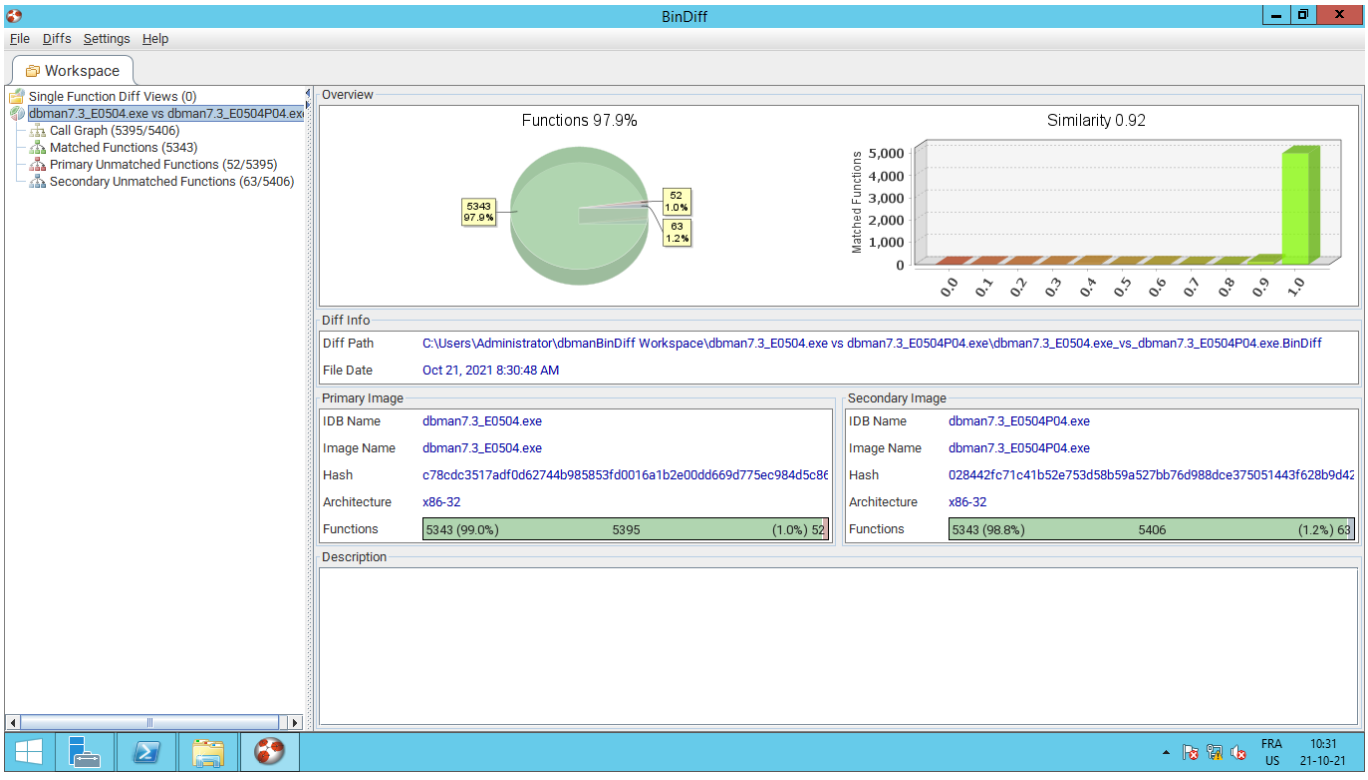


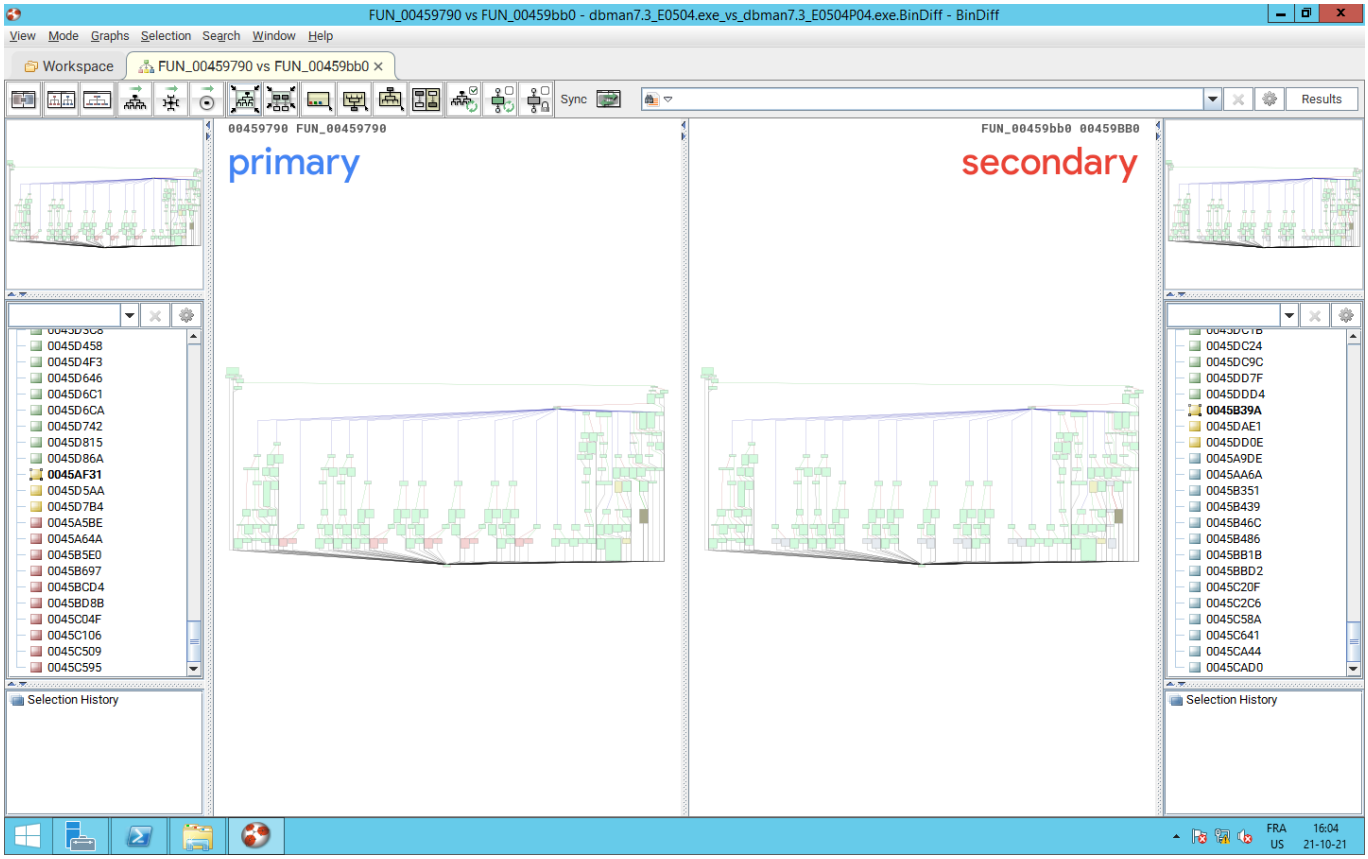
We can see:

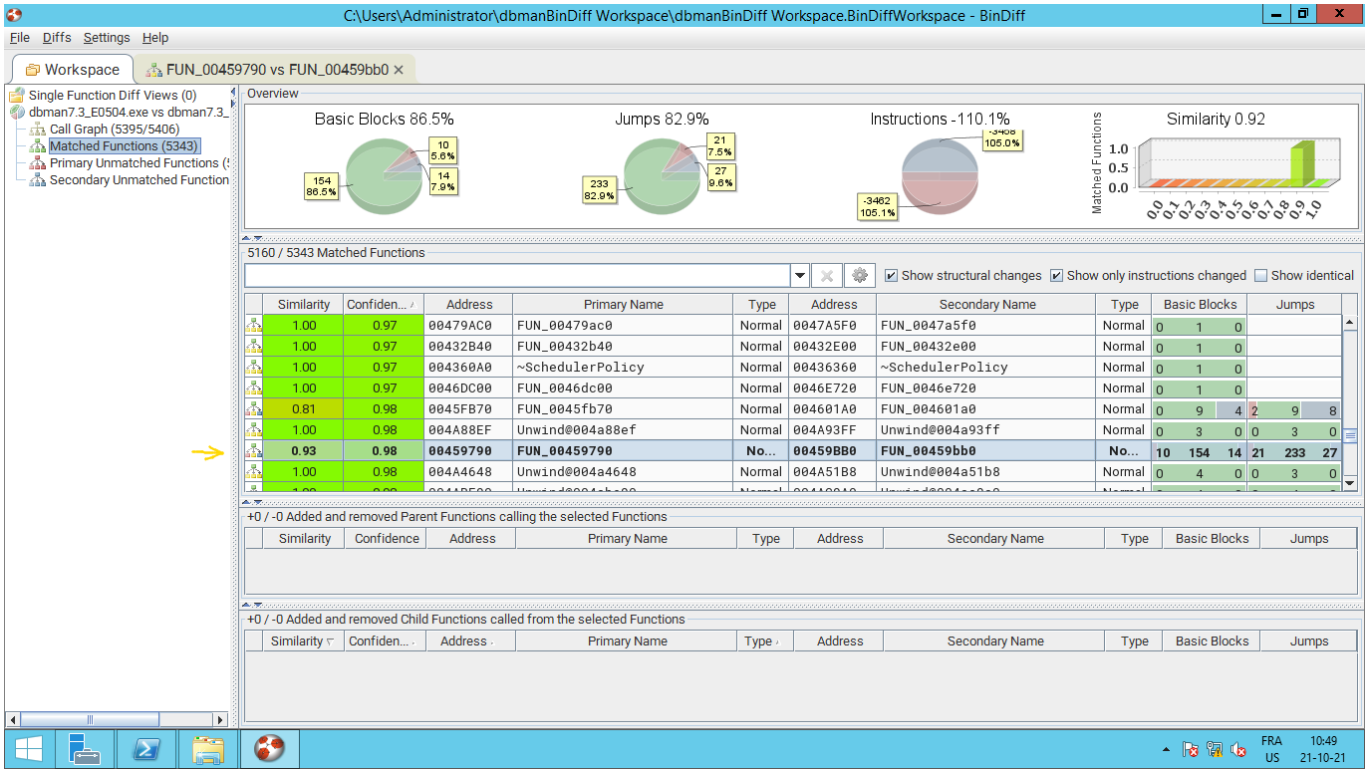
`FUN_00459790` (7.3_E0504) and its corresponding `FUN_00459bb0` (7.3_E0504P04):



Looking at this in **Bindiff7**:

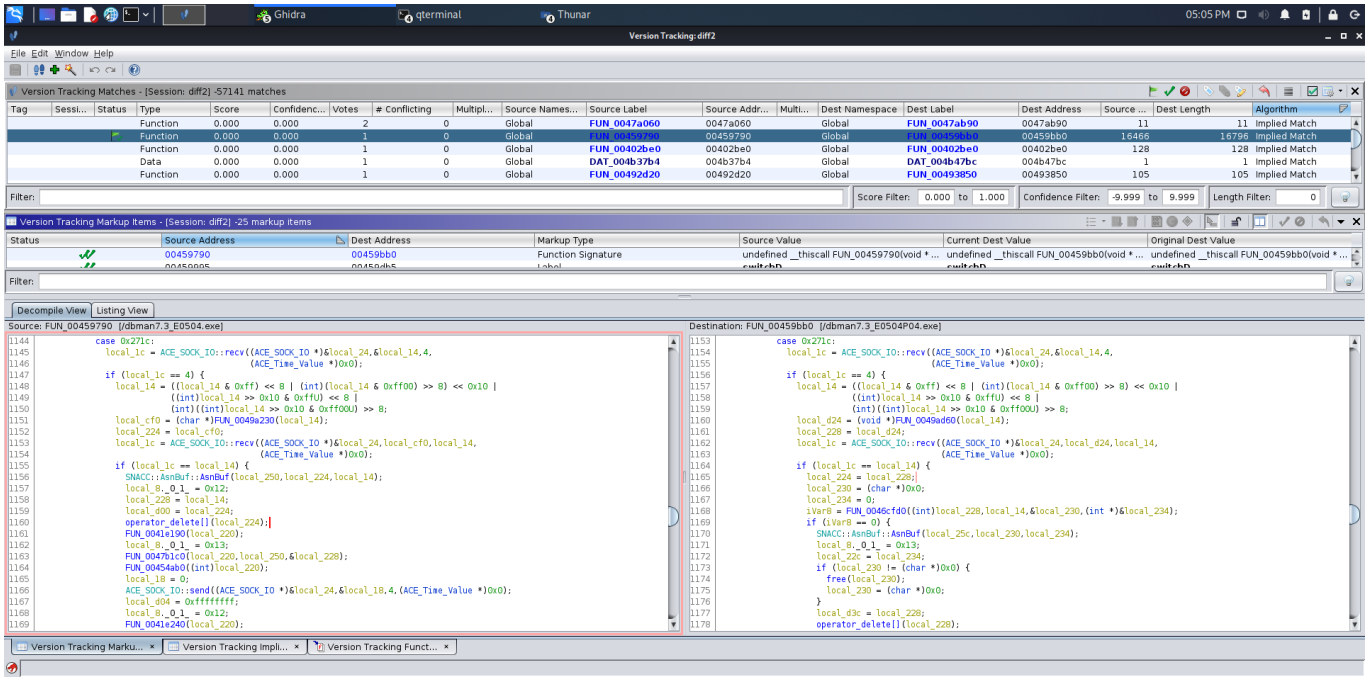






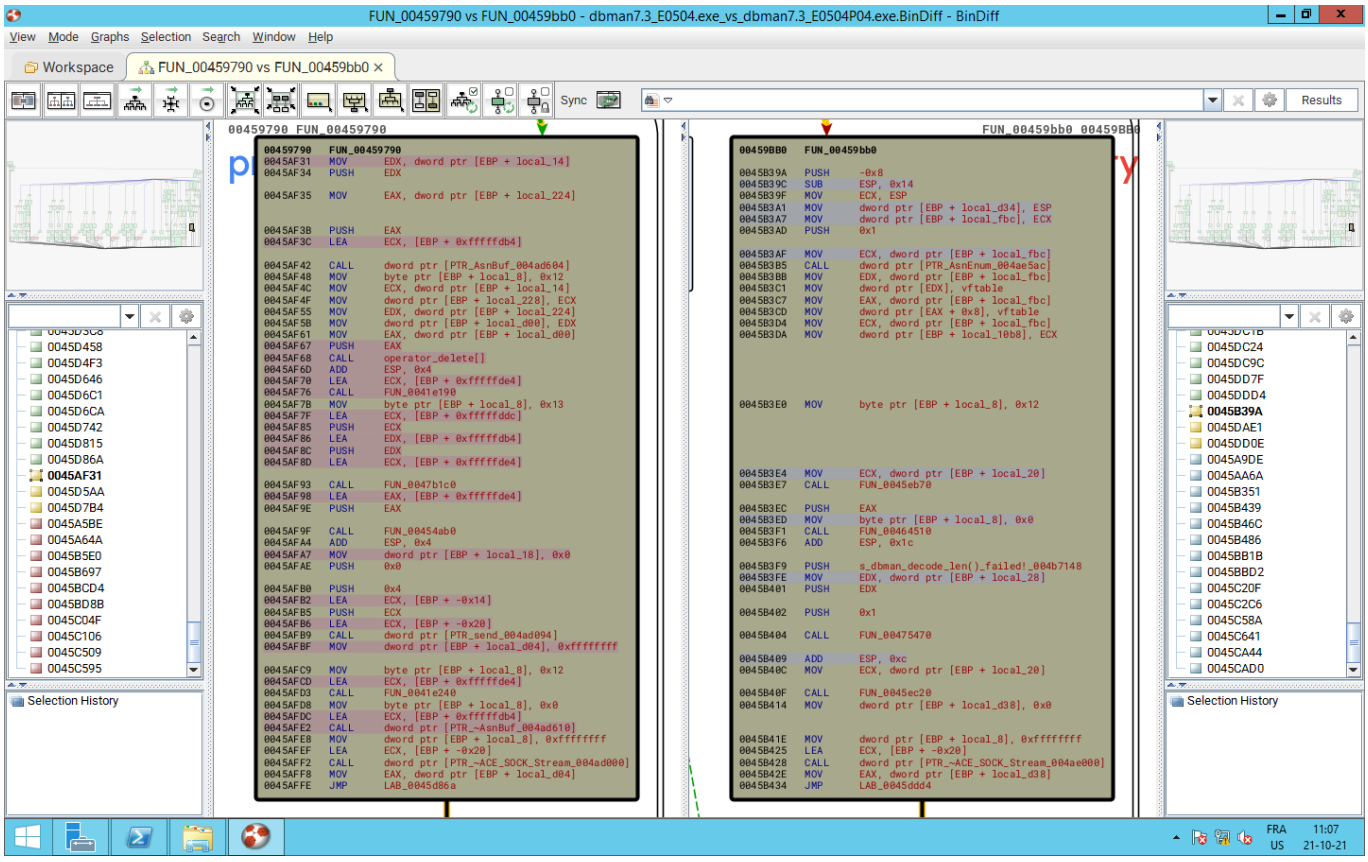
Back to Version Tracking:

Switch case for opcode 10012(`0x271c`):

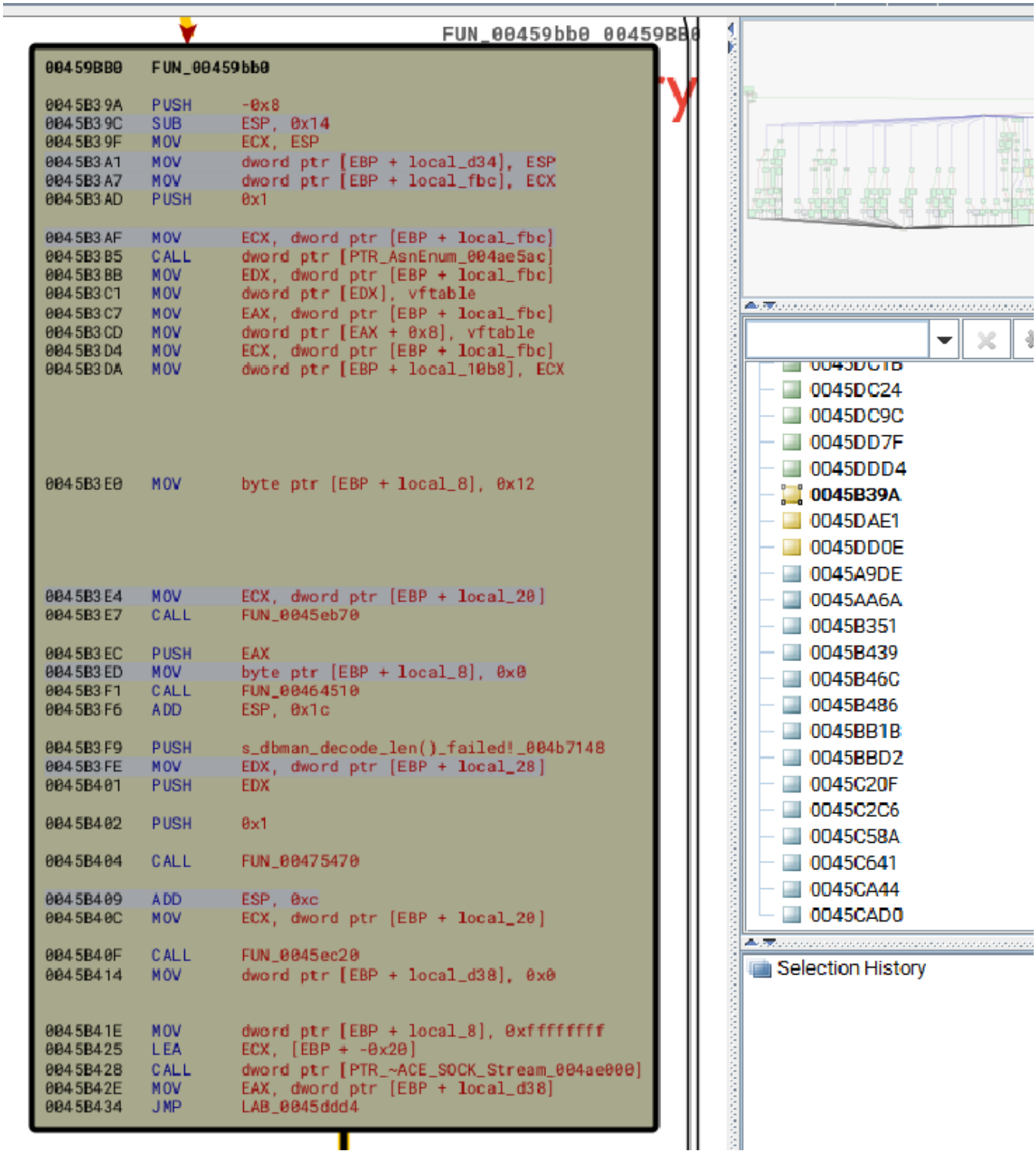


Bindiff:

`FUN_00459790` (7.3_E0504) and its corresponding `FUN_00459bb0` (7.3_E0504P04):



`FUN_00459bb0` (7.3_E0504P04):

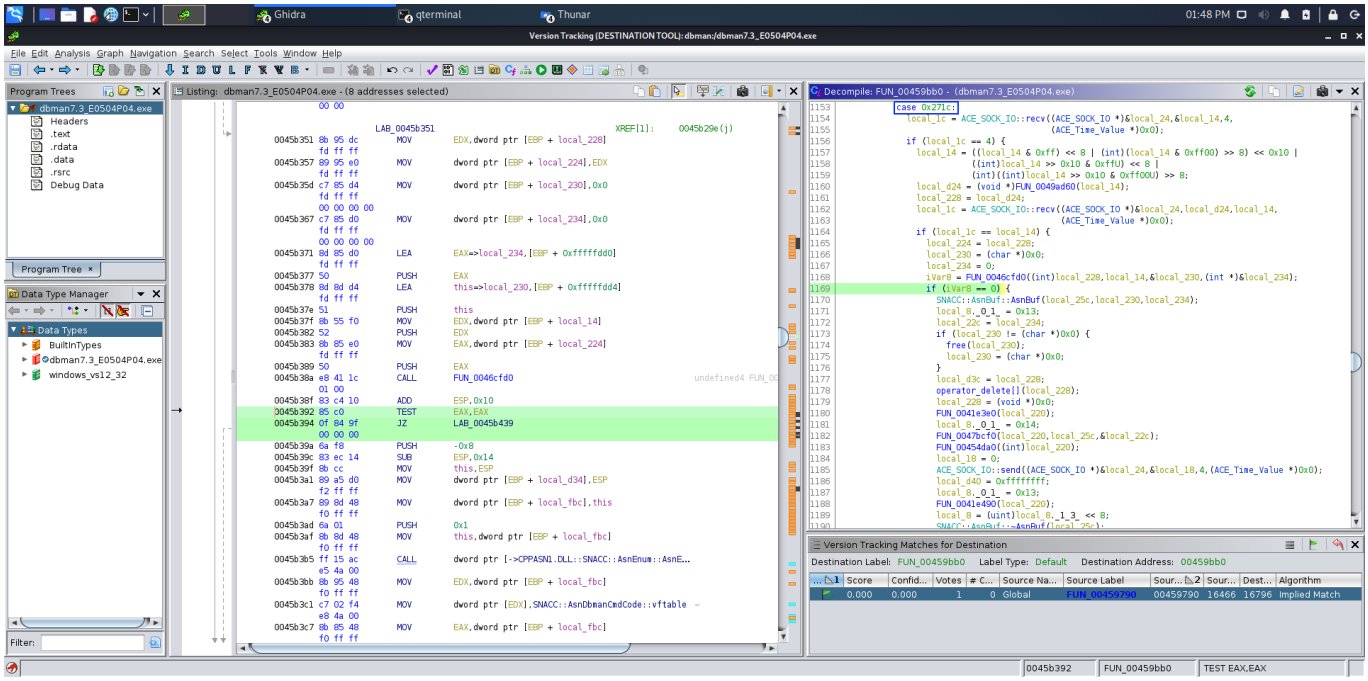


We are interested in:

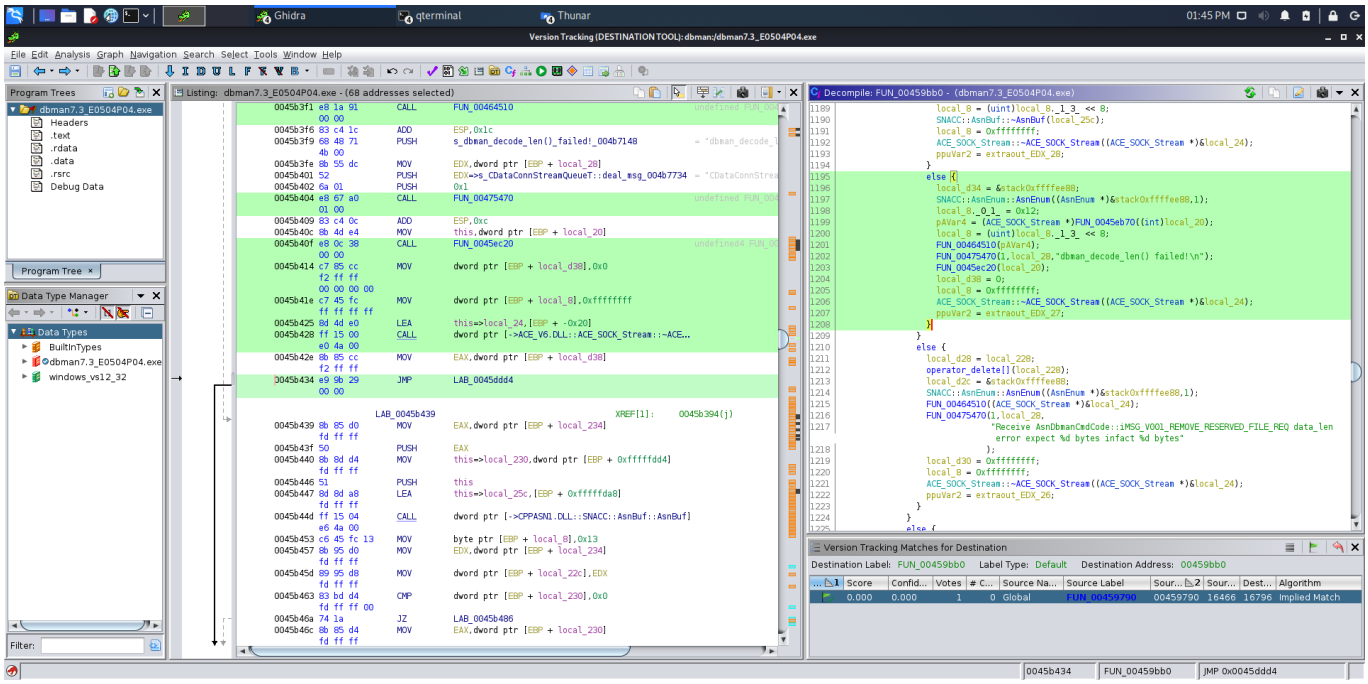
- @0045b40f: `FUN_0045ec20`

This is because of the following flow:

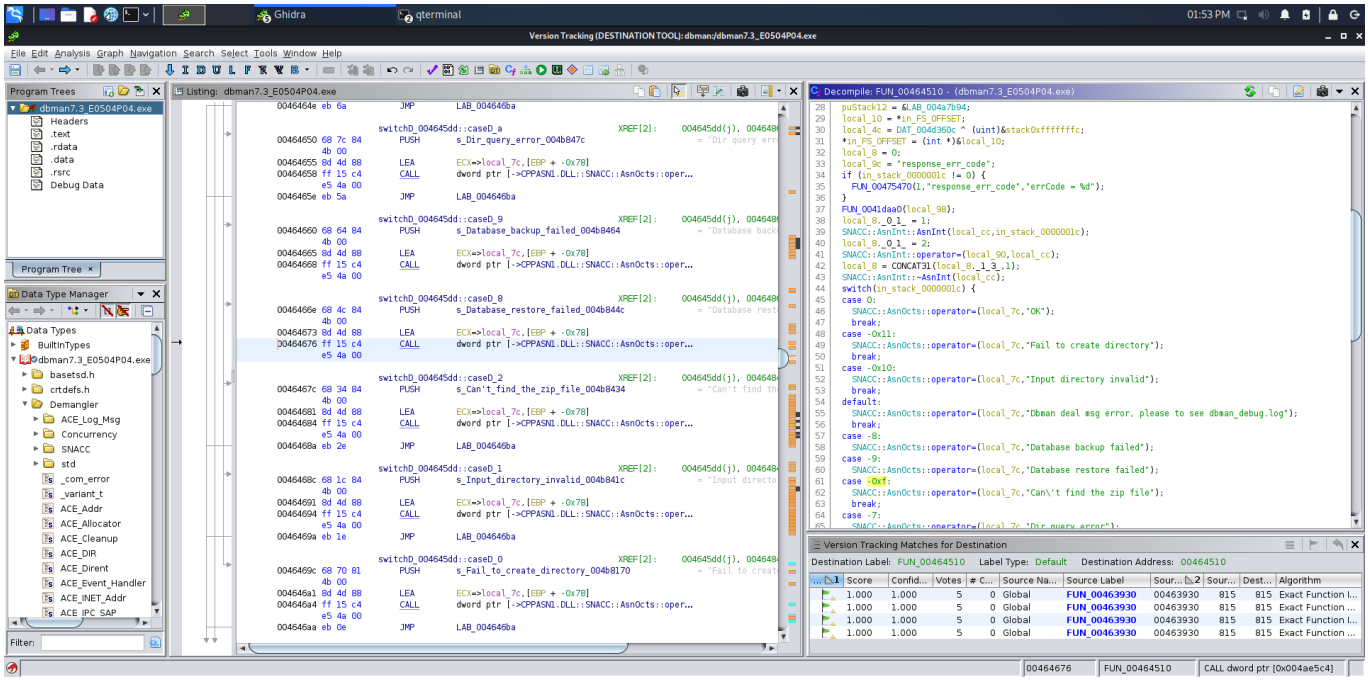
From/inside `FUN_00459bb0`(switch case for `0x271c`):



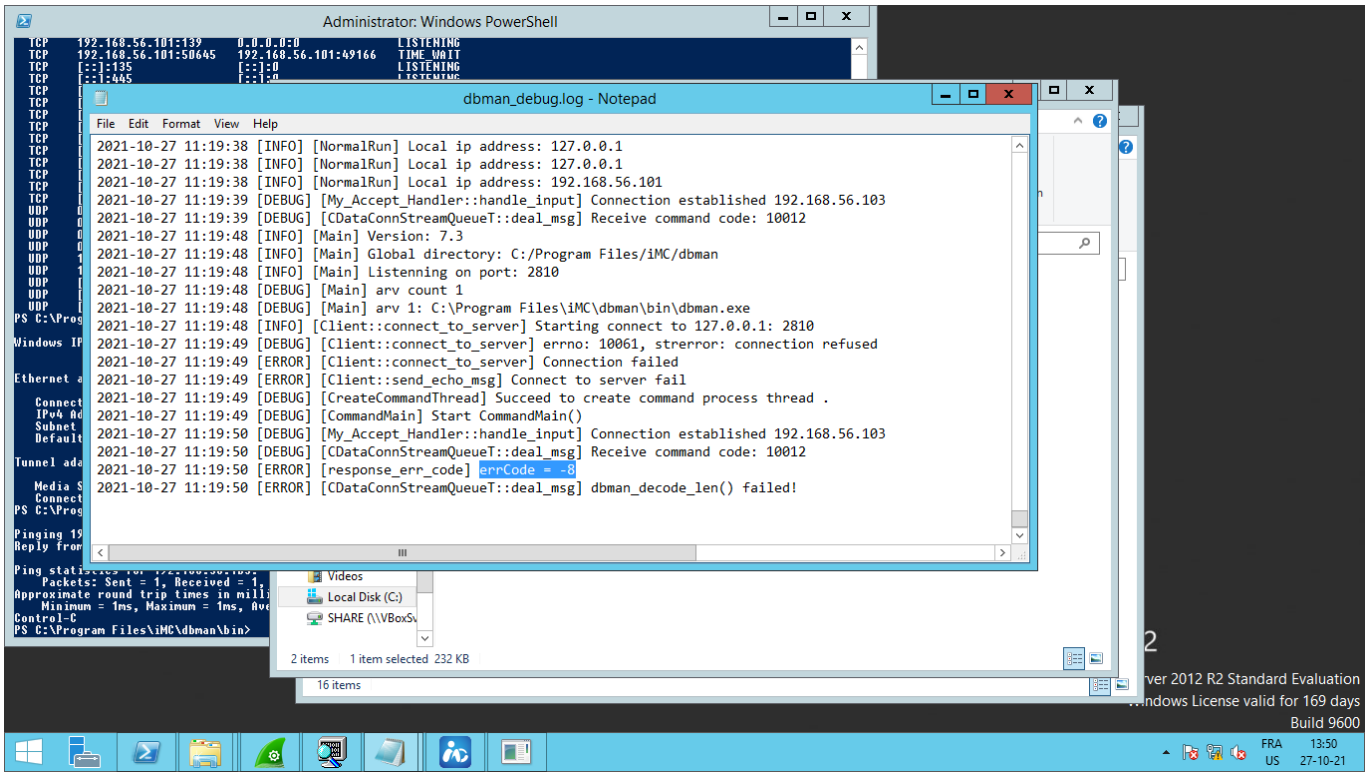
We need to fail this `if` condition to end up in the `else` clause:



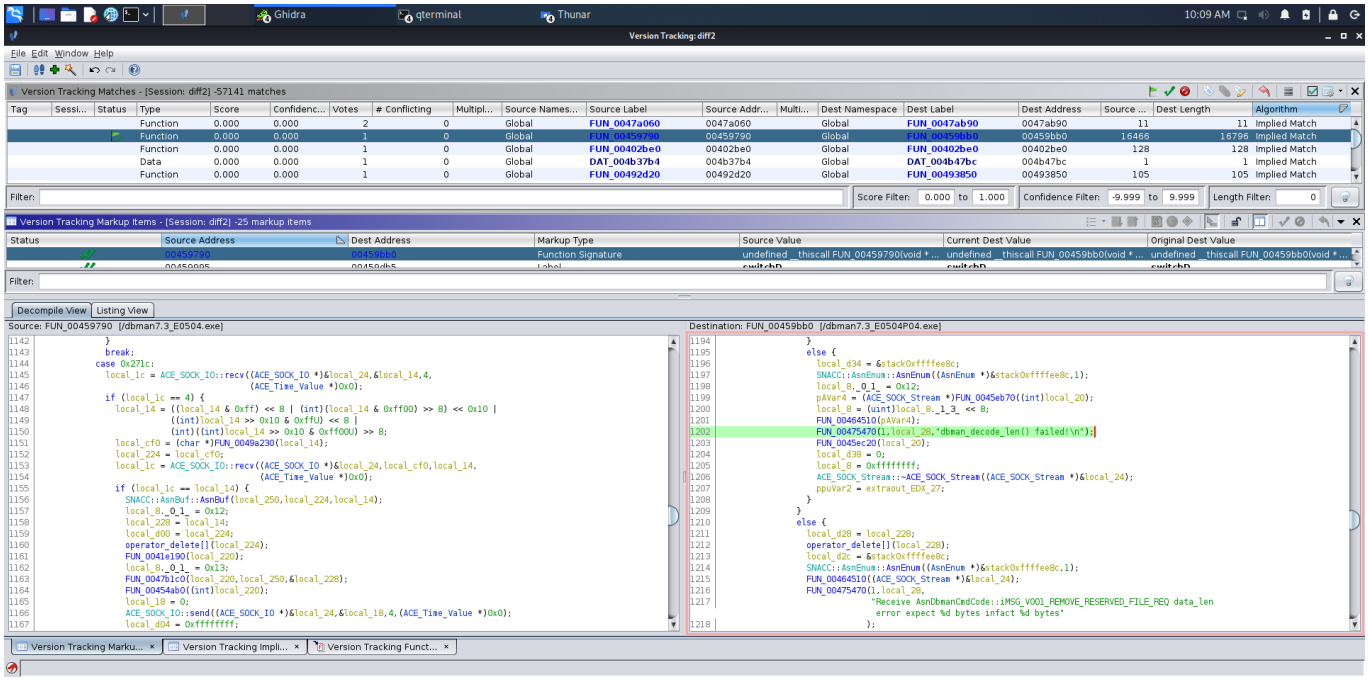
First, we enter `FUN_00464510`:



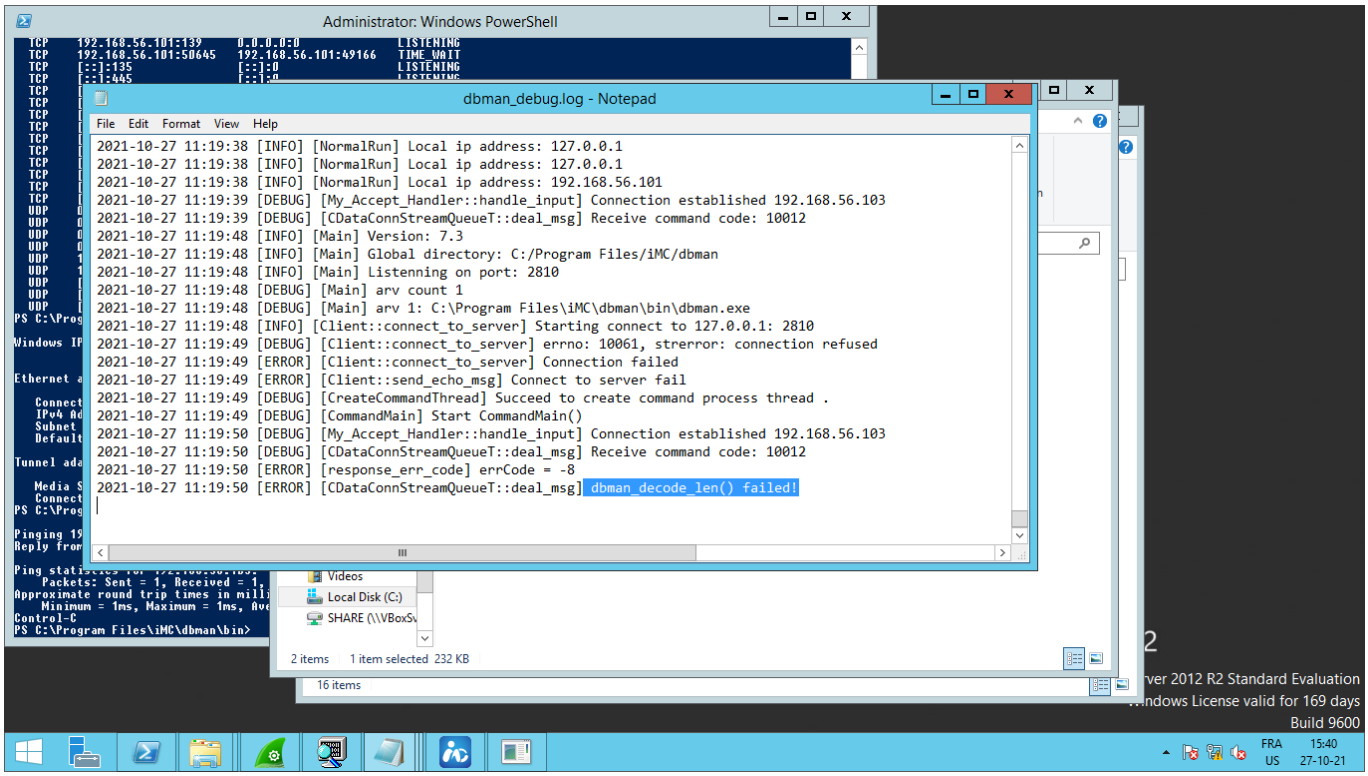
which writes to the dbman log file `errCode = -8` :



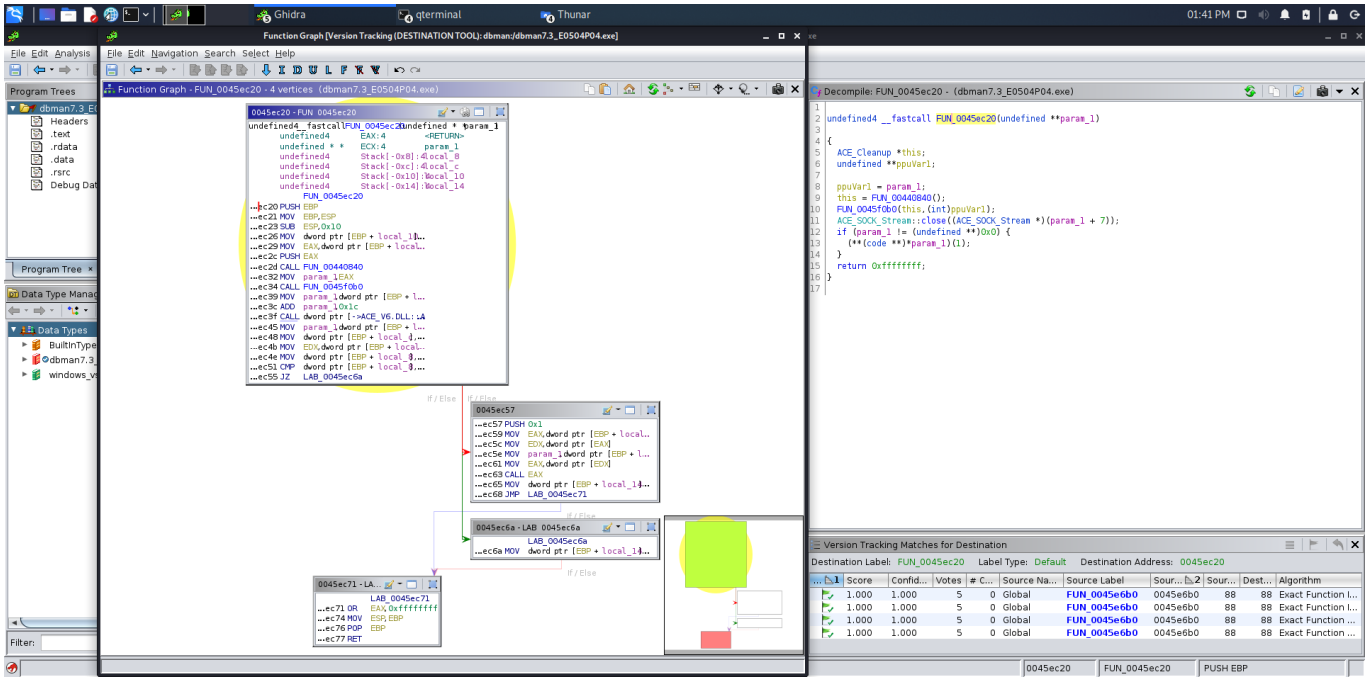
The call to `FUN_00475470` :



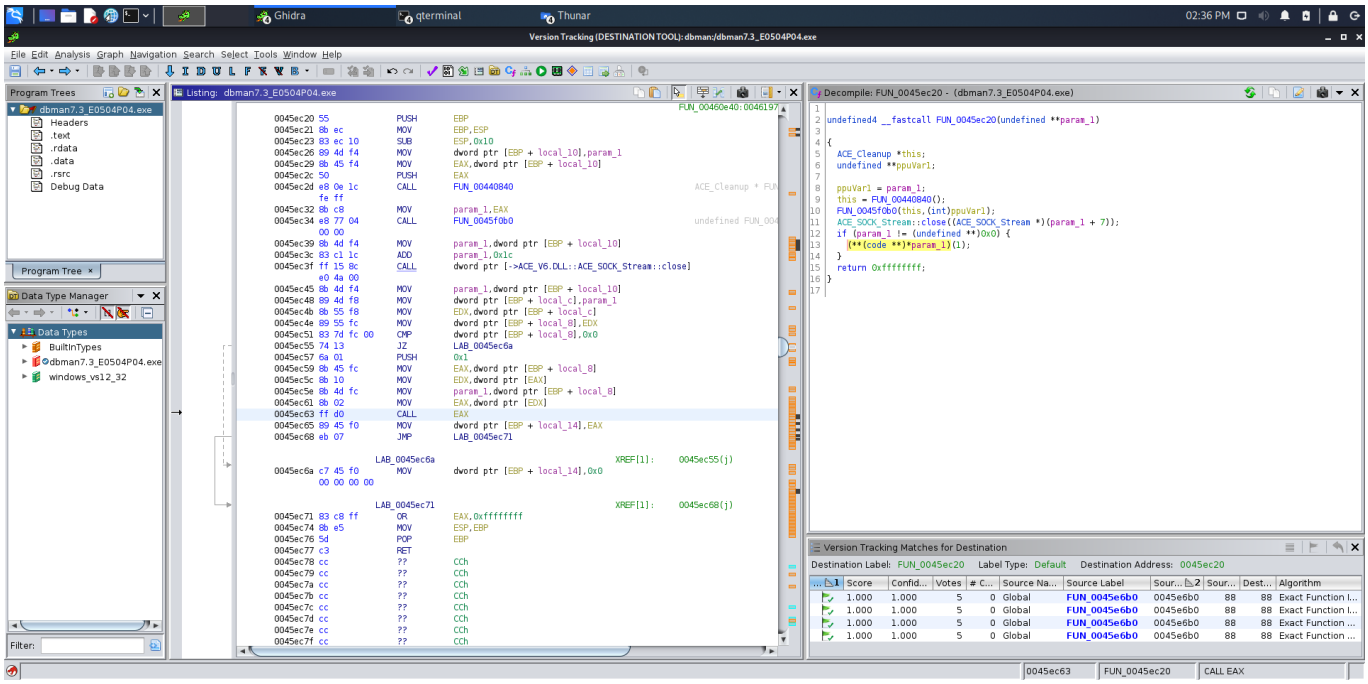
which writes `dbman_decode_len() failed` error to the log:



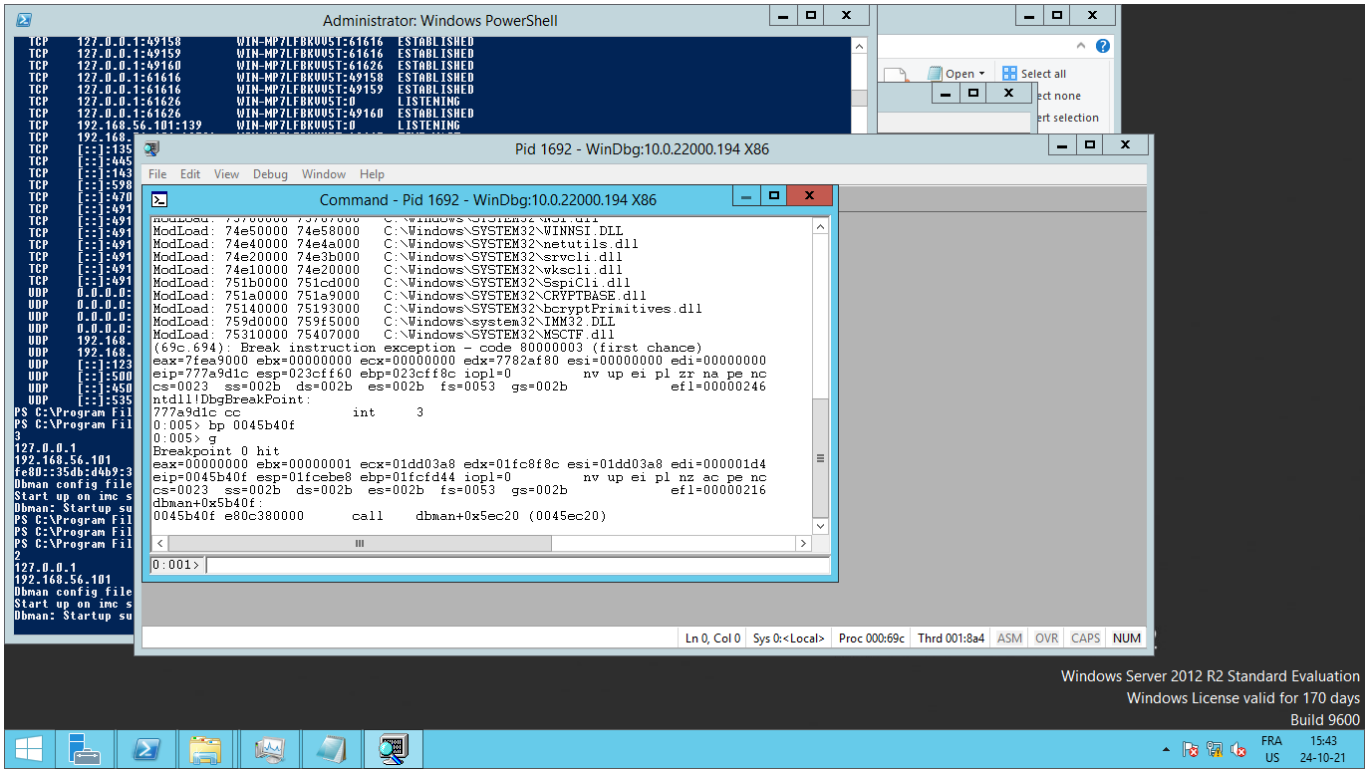
We're still in the `else` clause, we'll end up hitting `FUN_0045ec20` :



The `CALL EAX` needs to be resolved dynamically:

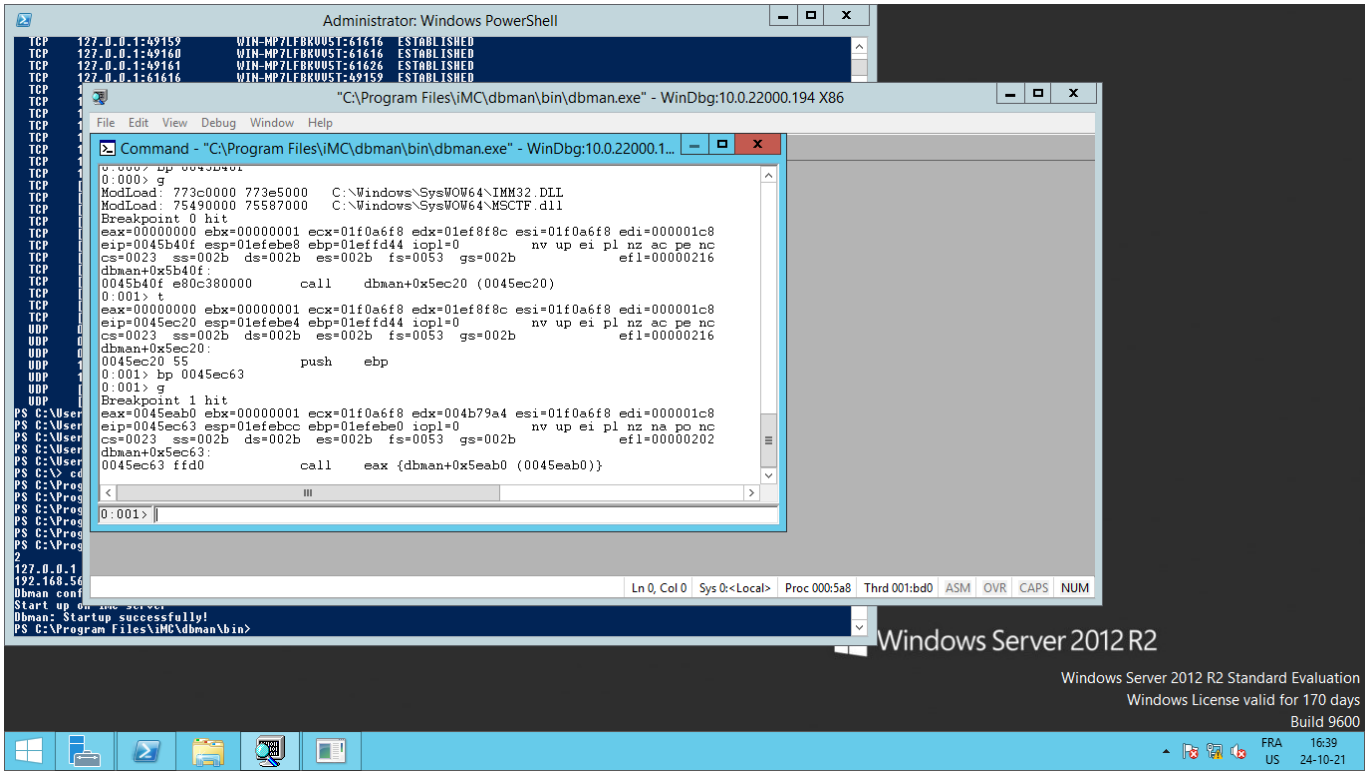


Looking/breakpoint for `FUN_0045ec20` at `0045b40f`:

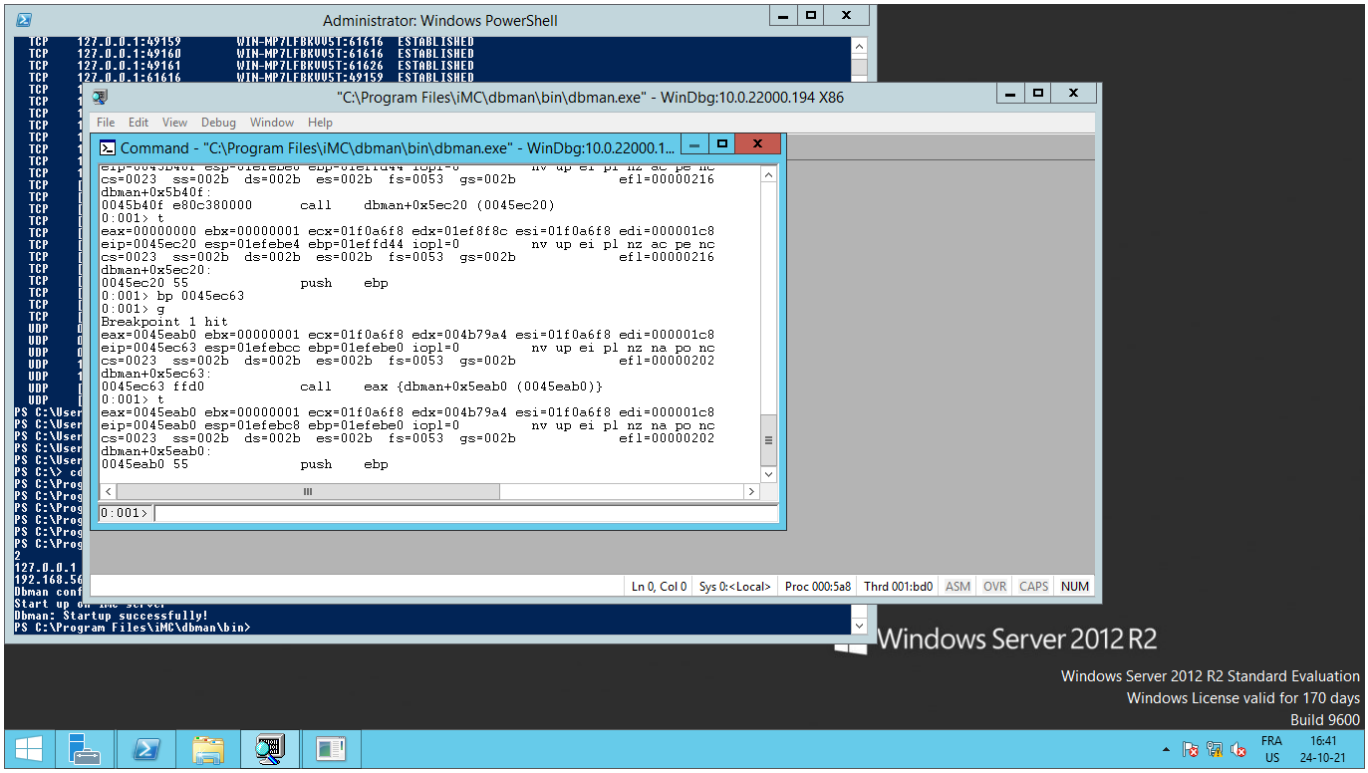


Next, we're looking for `CALL EAX` instruction which is located at: `0045ec63`.

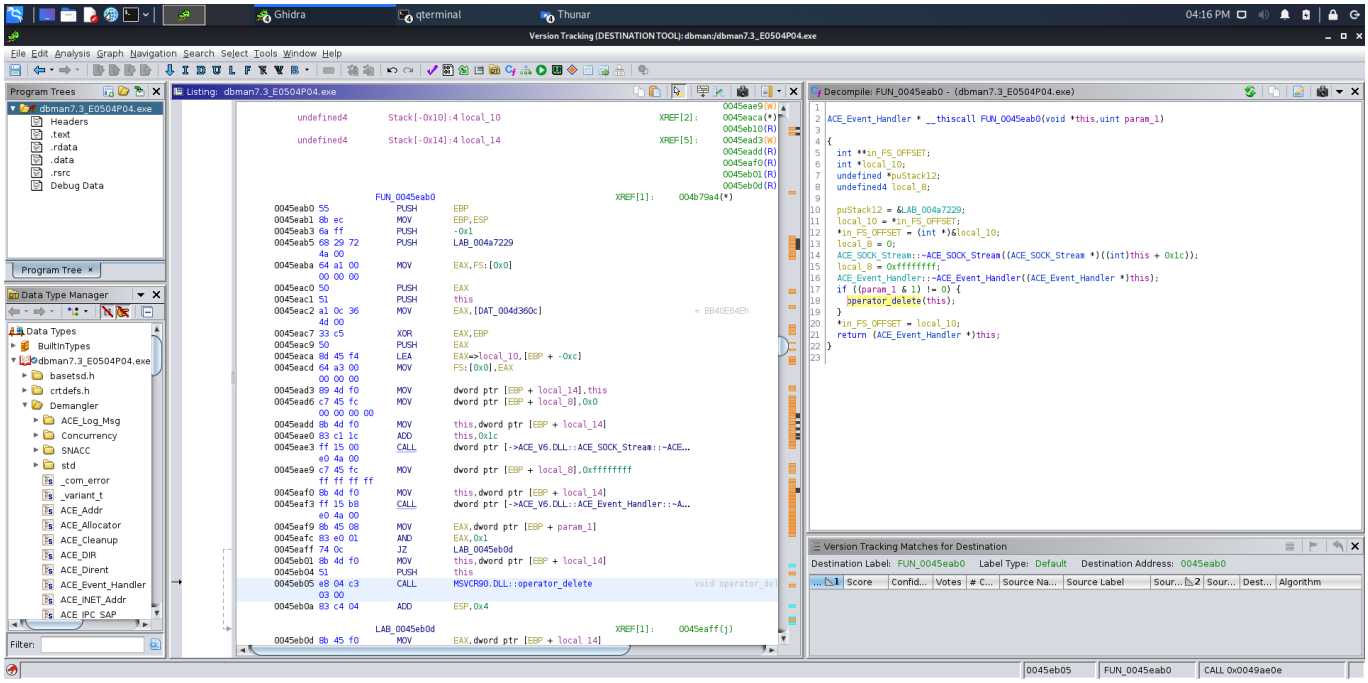
Looking at `0045ec63` from windbg:



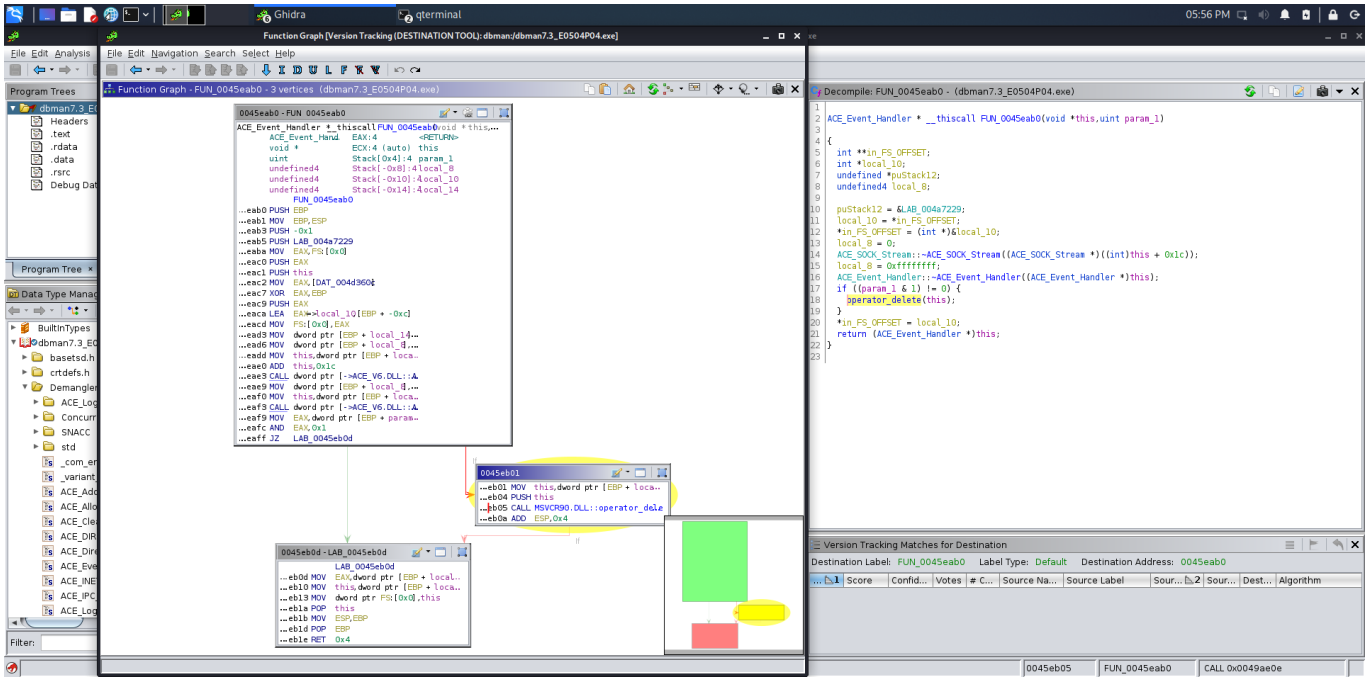
The next instruction after the *call* will resolve for `0045eab0`:



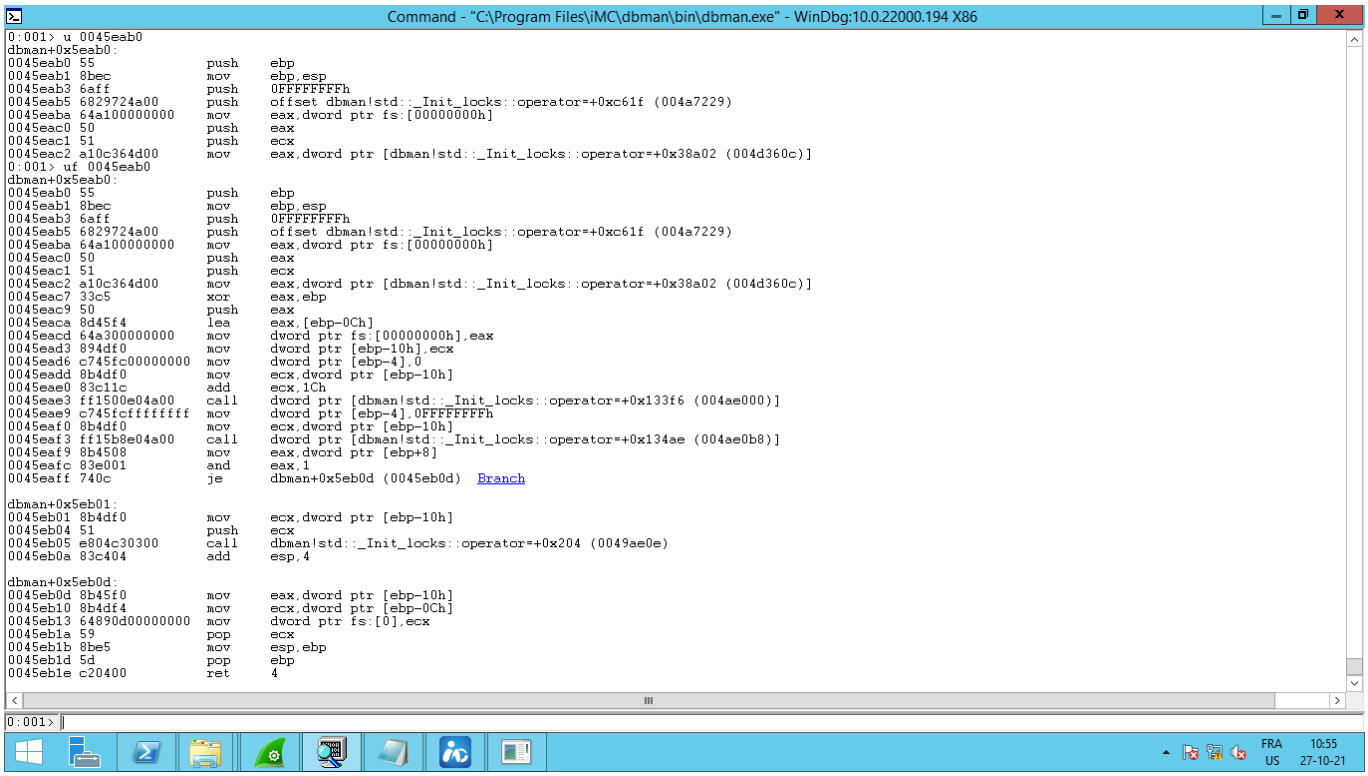
We see that it's calling `FUN_0045eab0`(where `operator_delete` is @0045eb05), which in Ghidra decompiler gives:



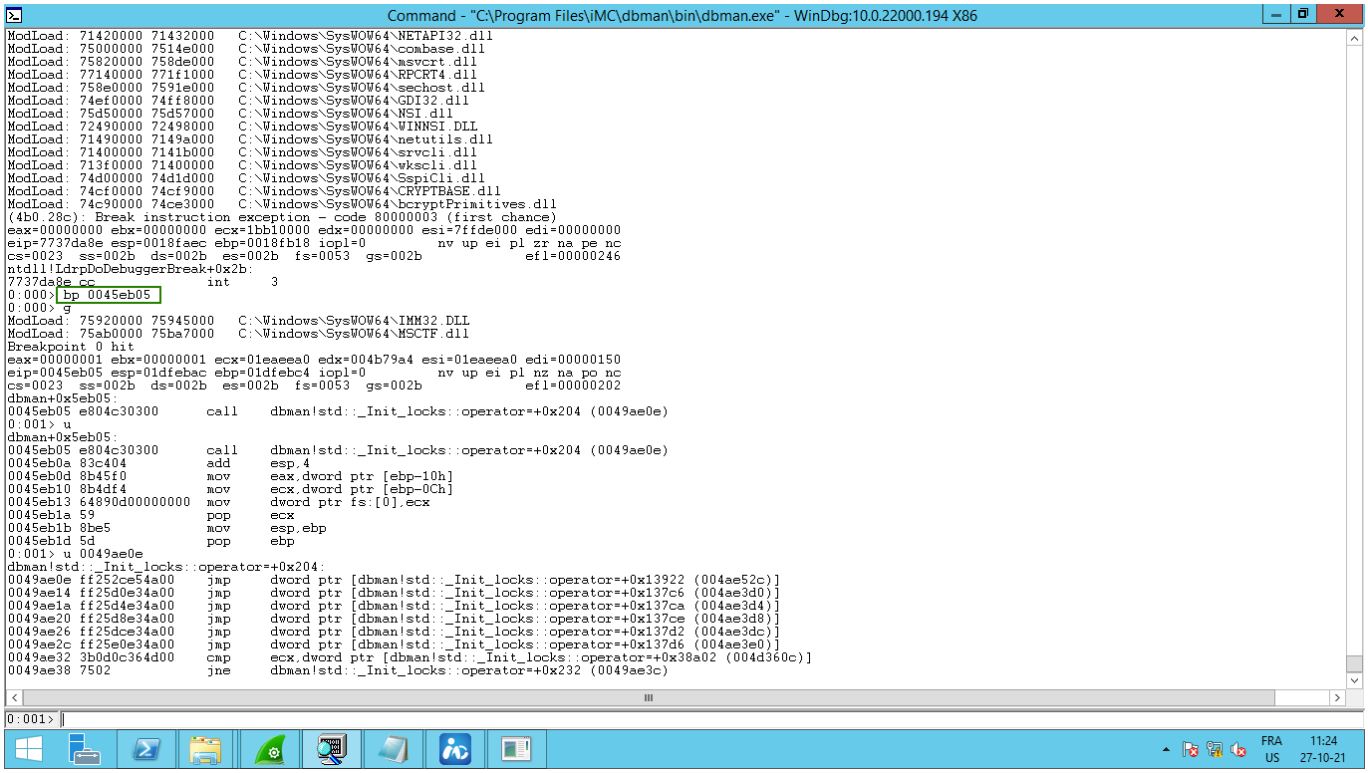
Graph view of it:



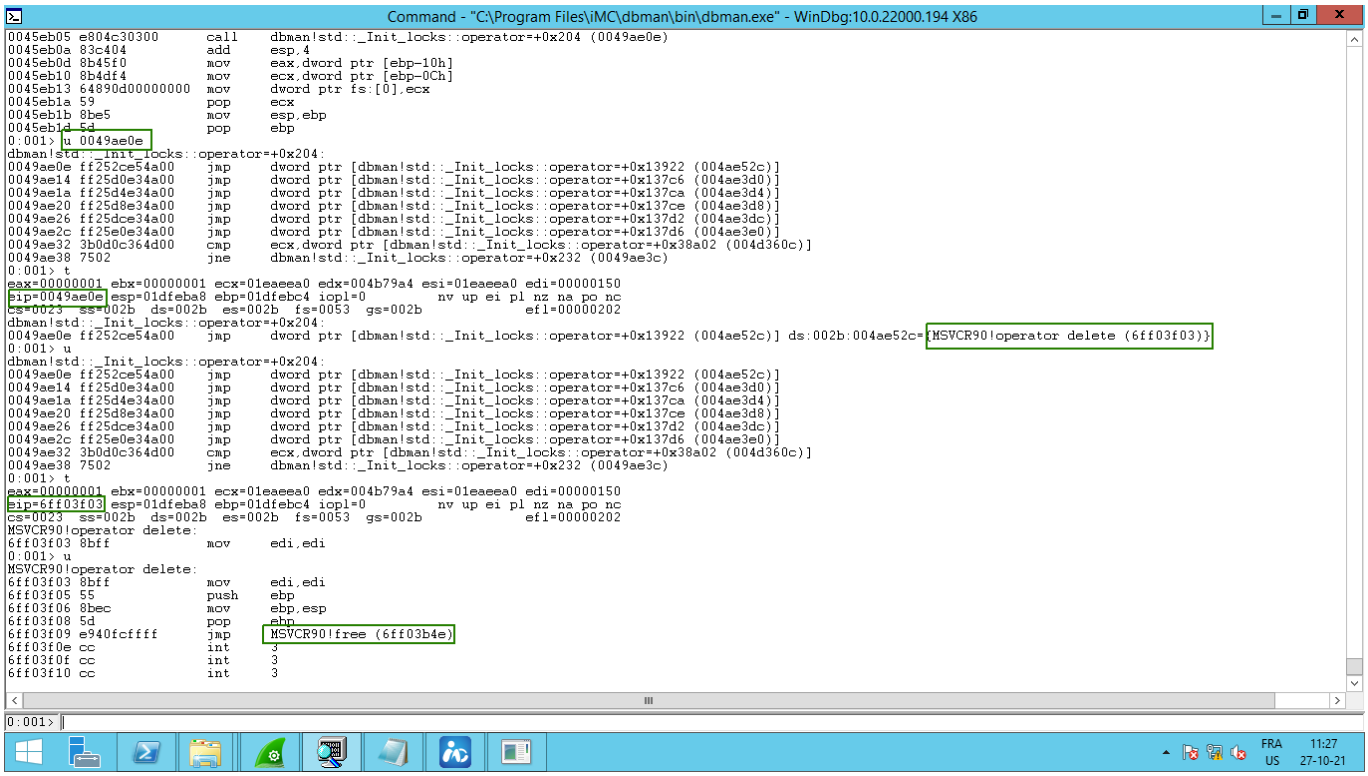
Looking at it(`FUN_0045eab0`) in windbg:



Windbg: breaking @0045eb05 (`operator_delete`):



It looks like `operator delete` makes use of `free`:



Analyzing through gflags.exe from WindowsSDK:

At times, tracing/following the binary's flow can be cumbersome ; especially for this type of bug.

Using the *gflags.exe* utility can help by putting objects into a full page heap, which can be found at:

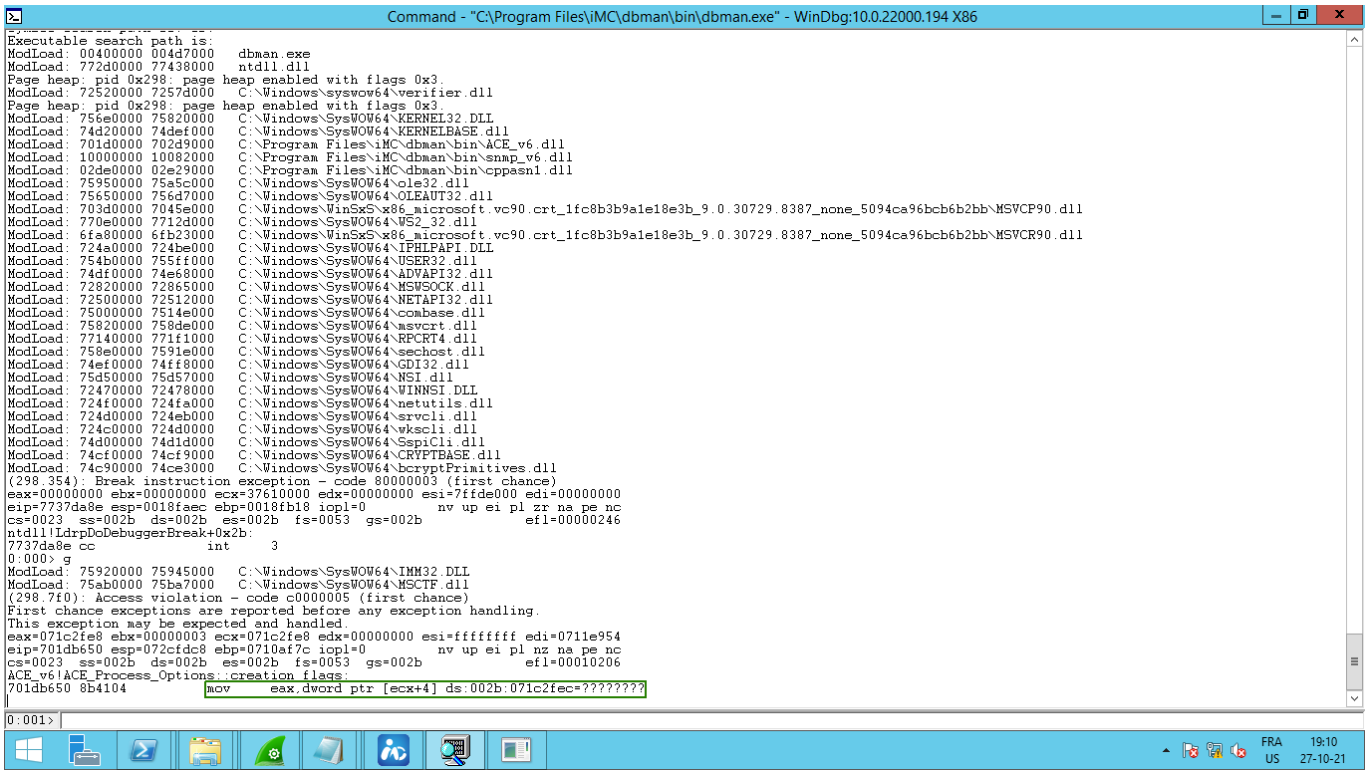
```
C:\Program Files (x86)\Windows Kits\10\Debuggers\x86\
```

Enabling *gflags* on *dbman*:

```
> .\gflags.exe /i "C:\Program Files\iMC\dbman\bin\dbman.exe" +hpa
```

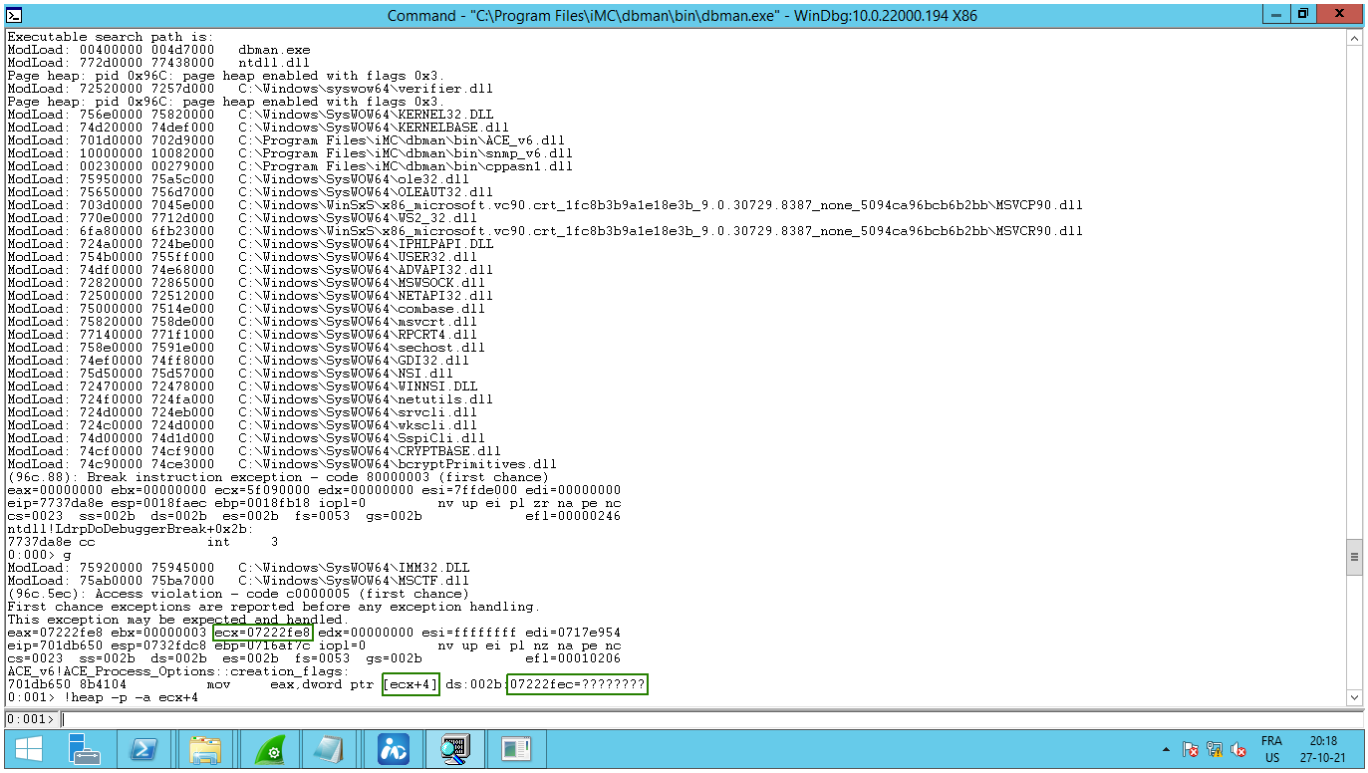
A bit like when you're trying to workout an exploit "inside" a debugger, this *dbman* behaviour isn't identical to the "vanilla" one.

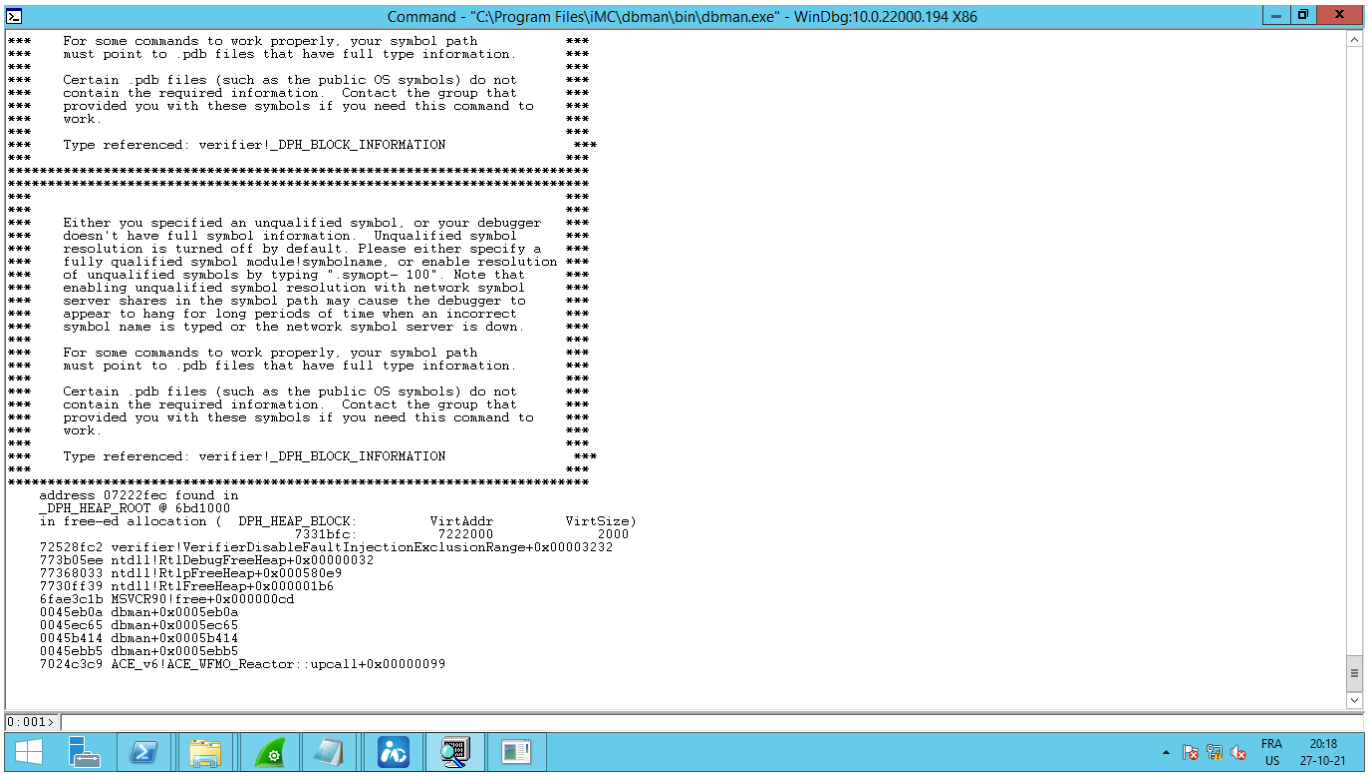
Observing the behaviour in windbg, *access violation*:



It breaks when it's trying to put the value pointed by `[ecx+4]` into `eax` register.

Looking at the heap:


Command - "C:\Program Files\iMC\dbman\bin\dbman.exe" - WinDbg:10.0.22000.194 X86
Executable search path is:
ModLoad: 00400000 004d7000   dbman.exe
ModLoad: 772d0000 77438000   ntdll.dll
Page heap: pid 0x96C: page heap enabled with flags 0x3.
ModLoad: 72520000 7257d000   C:\Windows\syswow64\verifier.dll
Page heap: pid 0x96C: page heap enabled with flags 0x3.
ModLoad: 756e0000 75820000   C:\Windows\SysWOW64\KERNEL32.DLL
ModLoad: 74d20000 74def000   C:\Windows\SysWOW64\KERNELBASE.dll
ModLoad: 701d0000 702d9000   C:\Program Files\iMC\dbman\bin\ACE_v6.dll
ModLoad: 10000000 10082000   C:\Program Files\iMC\dbman\bin\snmp_v6.dll
ModLoad: 00230000 00279000   C:\Program Files\iMC\dbman\bin\cppasn1.dll
ModLoad: 75950000 75a5c000   C:\Windows\SysWOW64\ole32.dll
ModLoad: 75650000 756d7000   C:\Windows\SysWOW64\OLEAUT32.dll
ModLoad: 703d0000 7045e000   C:\Windows\WinSxS\x86_microsoft.vc90.crt_1fc8b3b9a1e18e3b_9.0.30729.8387_none_5094ca96bcb6b2bb\MSVCP90.dll
ModLoad: 770e0000 7712d000   C:\Windows\SysWOW64\WS2_32.dll
ModLoad: 6fa80000 6fb23000   C:\Windows\WinSxS\x86_microsoft.vc90.crt_1fc8b3b9a1e18e3b_9.0.30729.8387_none_5094ca96bcb6b2bb\MSVCR90.dll
ModLoad: 724a0000 724be000   C:\Windows\SysWOW64\IPHLPAPI.DLL
ModLoad: 754b0000 755ff000   C:\Windows\SysWOW64\USER32.dll
ModLoad: 74df0000 74e68000   C:\Windows\SysWOW64\ADVAPI32.dll
ModLoad: 72820000 72865000   C:\Windows\SysWOW64\MSWSOCK.dll
ModLoad: 72500000 72512000   C:\Windows\SysWOW64\NETAPI32.dll
ModLoad: 75000000 7514e000   C:\Windows\SysWOW64\combase.dll
ModLoad: 75820000 758de000   C:\Windows\SysWOW64\msvcrt.dll
ModLoad: 77140000 771f1000   C:\Windows\SysWOW64\RPCRT4.dll
ModLoad: 758e0000 7591e000   C:\Windows\SysWOW64\sechost.dll
ModLoad: 74ef0000 74ff8000   C:\Windows\SysWOW64\GDI32.dll
ModLoad: 75d50000 75d57000   C:\Windows\SysWOW64\NSI.dll
ModLoad: 72470000 72478000   C:\Windows\SysWOW64\WINNSI.DLL
ModLoad: 724f0000 724fa000   C:\Windows\SysWOW64\netutils.dll
ModLoad: 724d0000 724eb000   C:\Windows\SysWOW64\srvcli.dll
ModLoad: 724c0000 724d0000   C:\Windows\SysWOW64\wkscli.dll
ModLoad: 74d00000 74d1d000   C:\Windows\SysWOW64\SspiCli.dll
ModLoad: 74cf0000 74cf9000   C:\Windows\SysWOW64\CRYPTBASE.dll
ModLoad: 74c90000 74ce3000   C:\Windows\SysWOW64\bcryptPrimitives.dll
(96c.88): Break instruction exception - code 80000003 (first chance)
eax=00000000 ebx=00000000 ecx=5f090000 edx=00000000 esi=7ffde000 edi=00000000
eip=7737da8e esp=0018faec ebp=0018fb18 iopl=0   nv up ei pl zr na pe nc
cs=0023 ss=002b ds=002b es=002b fs=0053 gs=002b   efl=00000246
ntdll!LdrpDoDebuggerBreak+0x2b:
7737da8e cc   int   3
0:000> g
ModLoad: 75920000 75945000   C:\Windows\SysWOW64\IMM32.DLL
ModLoad: 75ab0000 75ba7000   C:\Windows\SysWOW64\MSCTF.dll
(96c.5ec): Access violation - code c0000005 (first chance)
First chance exceptions are reported before any exception handling.
This exception may be expected and handled.
eax=07222fe8 ebx=00000003 ecx=07222fe8 edx=00000000 esi=ffffffff edi=0717e954
eip=701db650 esp=0732fdc8 ebp=0716af7c iopl=0   nv up ei pl nz na pe nc
cs=0023 ss=002b ds=002b es=002b fs=0053 gs=002b   efl=00010206
ACE_v6!ACE_Process_Options::creation_flags:
701db650 8b4104   mov   eax,dword ptr [ecx+4] ds:002b:07222fec=????????
0:001> !heap -p -a ecx+4
0:001>

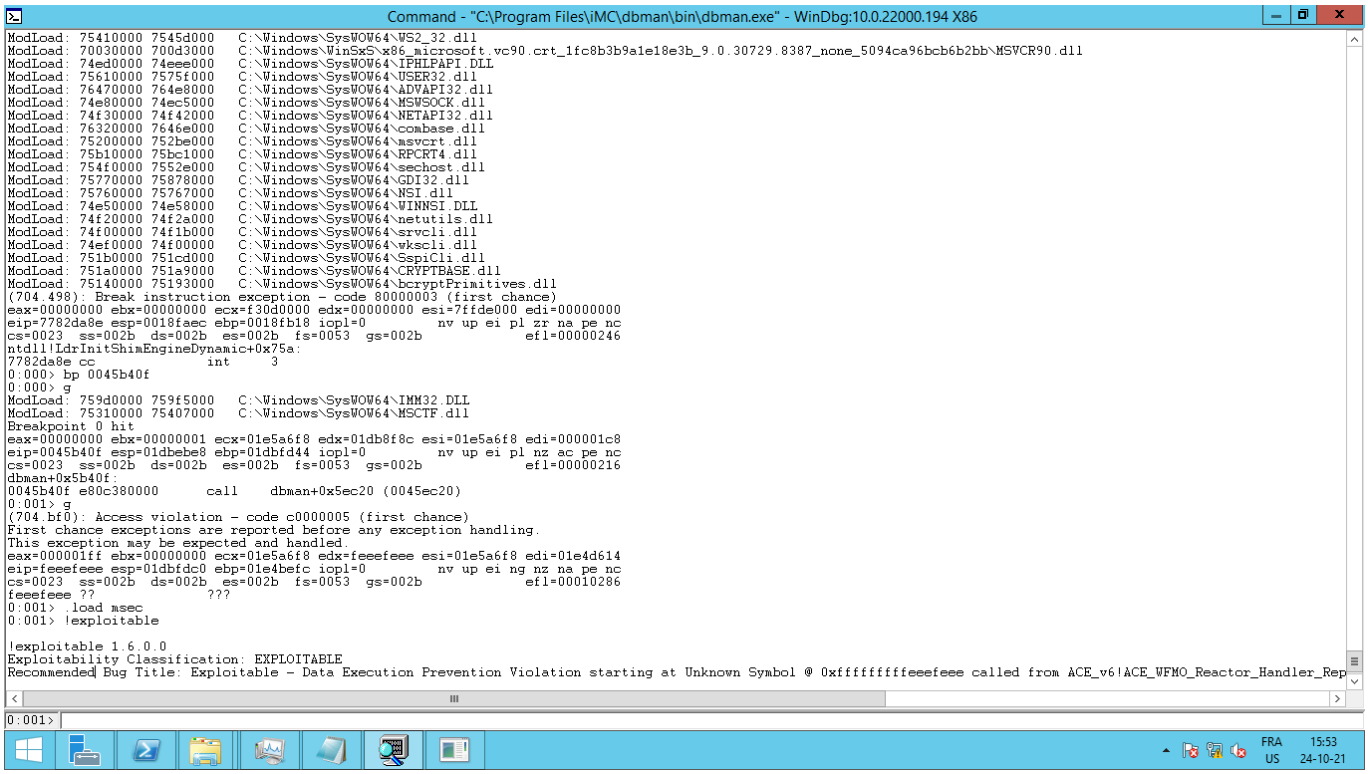

Command - "C:\Program Files\iMC\dbman\bin\dbman.exe" - WinDbg:10.0.22000.194 X86
***   For some commands to work properly, your symbol path   ***
***   must point to .pdb files that have full type information.   ***
***   ***
***   Certain .pdb files (such as the public OS symbols) do not   ***
***   contain the required information. Contact the group that   ***
***   provided you with these symbols if you need this command to   ***
***   work.   ***
***   ***
***   Type referenced: verifier!_DPH_BLOCK_INFORMATION   ***
***   ***
****
****
***   ***
***   ***
***   Either you specified an unqualified symbol, or your debugger   ***
***   doesn't have full symbol information. Unqualified symbol   ***
***   resolution is turned off by default. Please either specify a   ***
***   fully qualified symbol module!symbolname, or enable resolution ***
***   of unqualified symbols by typing ".symopt- 100". Note that   ***
***   enabling unqualified symbol resolution with network symbol   ***
***   server shares in the symbol path may cause the debugger to   ***
***   appear to hang for long periods of time when an incorrect   ***
***   symbol name is typed or the network symbol server is down.   ***
***   ***
***   For some commands to work properly, your symbol path   ***
***   must point to .pdb files that have full type information.   ***
***   ***
***   Certain .pdb files (such as the public OS symbols) do not   ***
***   contain the required information. Contact the group that   ***
***   provided you with these symbols if you need this command to   ***
***   work.   ***
***   ***
***   Type referenced: verifier!_DPH_BLOCK_INFORMATION   ***
***   ***
****
    address 07222fec found in
    _DPH_HEAP_ROOT @ 6bd1000
    in free-ed allocation (DPH_HEAP_BLOCK: VirtAddr   VirtSize)
                                 7331bfc: 7222000   2000
    72528fc2 verifier!VerifierDisableFaultInjectionExclusionRange+0x00003232
    773b05ee ntdll!RtlDebugFreeHeap+0x00000032
    77368033 ntdll!RtlpFreeHeap+0x000580e9
    7730ff39 ntdll!RtlFreeHeap+0x000001b6
    6fae3c1b MSVCR90!free+0x000000cd
    0045eb0a dbman+0x0005eb0a
    0045ec65 dbman+0x0005ec65
    0045b414 dbman+0x0005b414
    0045ebb5 dbman+0x0005ebb5
    7024c3c9 ACE_v6!ACE_WFMO_Reactor::upcall+0x00000099
0:001>

```
0:001> d ecx+4
07222fec  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????
07222ffc  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????
0722300c  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????
0722301c  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????
0722302c  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????
0722303c  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????
0722304c  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????
0722305c  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????
0:001> d eax
07222fe8  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????
07222ff8  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????
07223008  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????
07223018  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????
07223028  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????
07223038  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????
07223048  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????
07223058  ?? ?? ?? ?? ?? ?? ?? ??-?? ?? ?? ?? ?? ?? ?? ??  ????????????????

0:001>
```

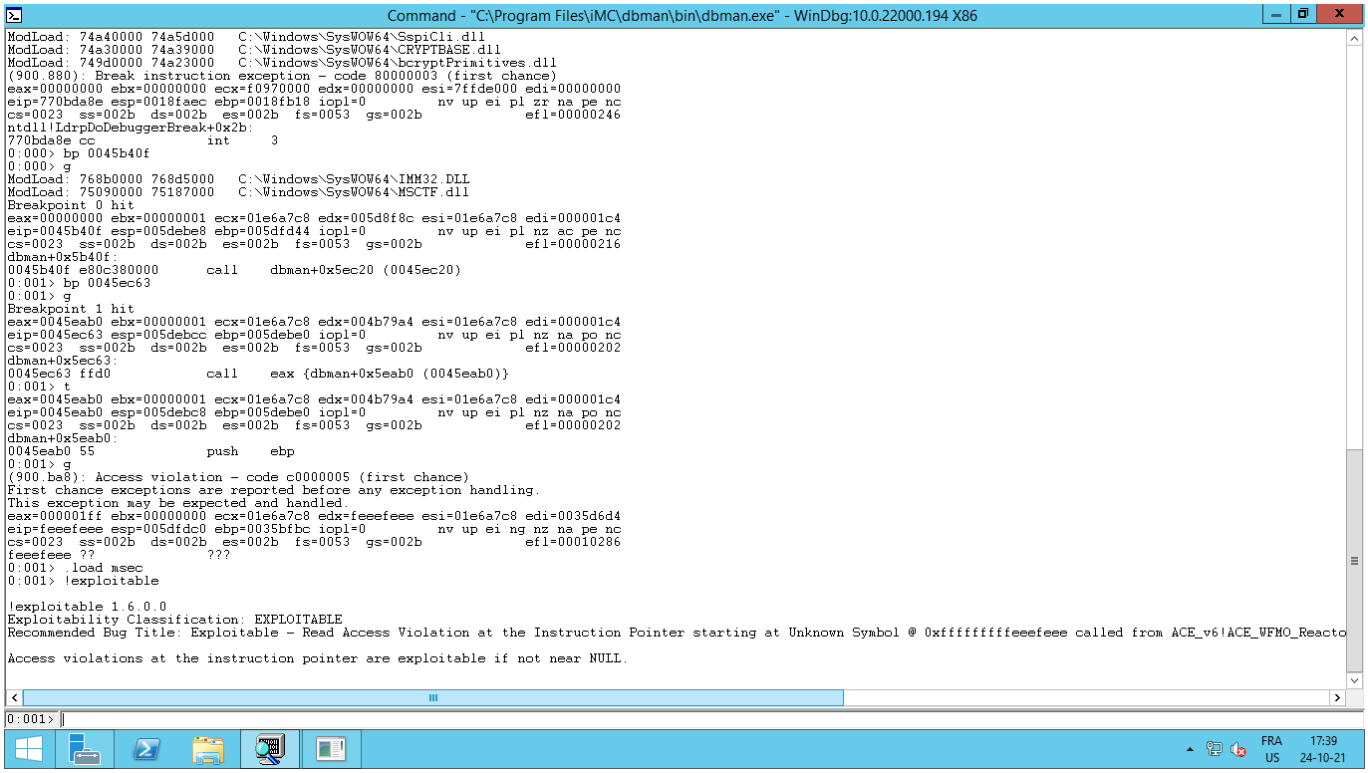
`[ecx+4]` register address belongs to the heap block that has been freed. By trying to "move"/dereference that value into `eax`, it triggers an *access violation*(AV).

AV code in windbg:

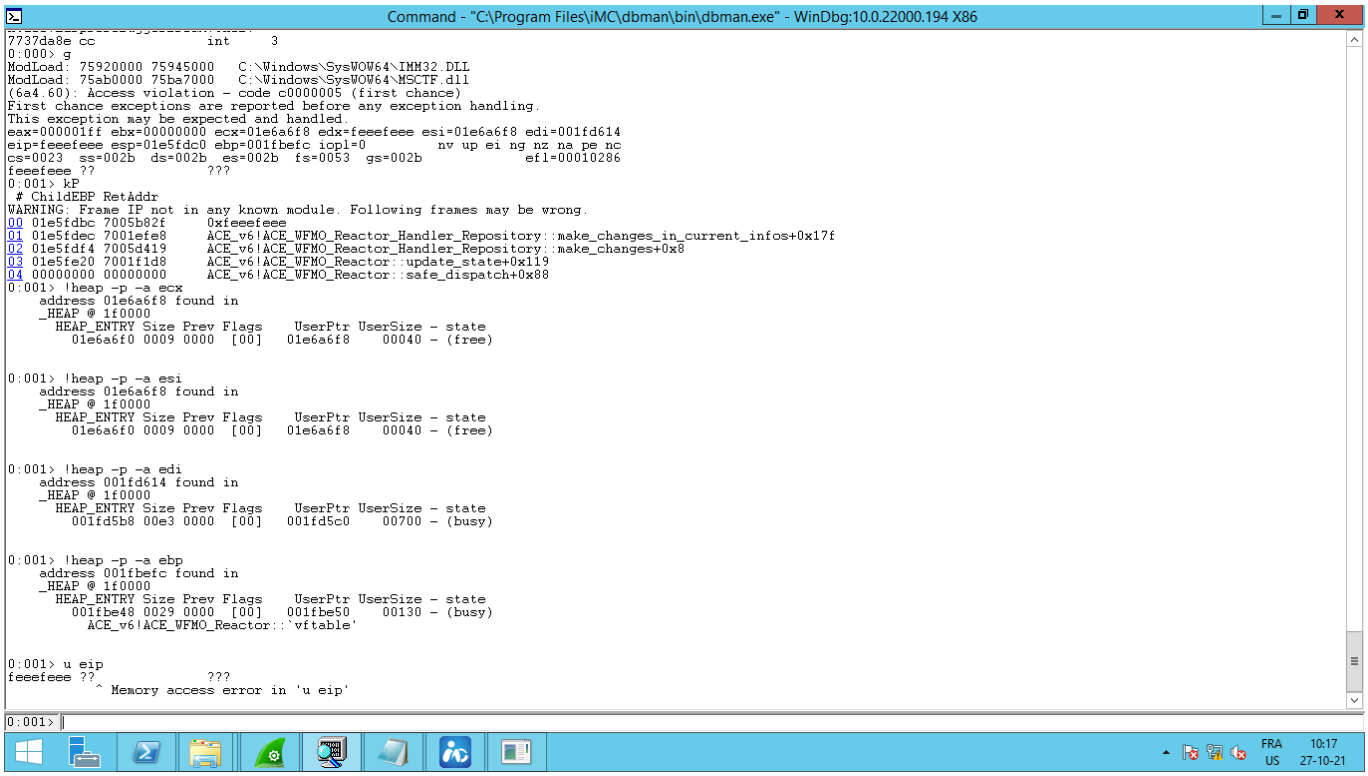
Access violation - code `c0000005` with DEP enabled:

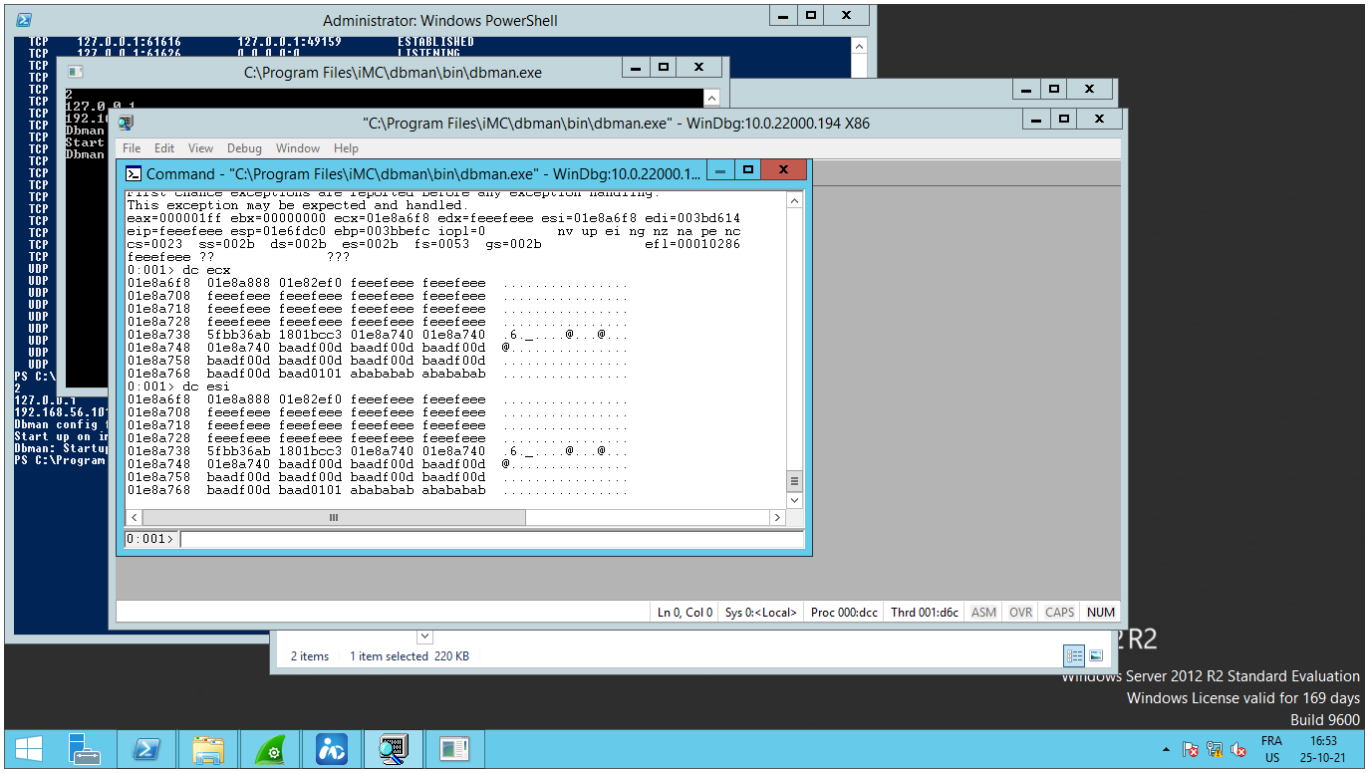


Access violation - code `c0000005` without DEP:



Looking at register/heap just after AV:





From: https://en.wikipedia.org/wiki/Magic_number_(programming)

`BAADF00D`:

BAADF00D	*"Bad food"*, Used by Microsoft's debug HeapAlloc() to mark uninitialized allocated heap memory[23]

`FEEEFEEE`:

FEEEFEEE	*"Fee fee"*, Used by Microsoft's debug HeapFree() to mark freed heap memory. Some nearby internal bookkeeping values may have the high word set to FEEE as well.[23]

These, confirm that there is some heap magic going on.

Refining analysis:

Loading `dbman.exe` from windbg:

```
ModLoad: 75520000 75529000   C:\Windows\SysWOW64\CRYPTBASE.dll
ModLoad: 754c0000 75513000   C:\Windows\SysWOW64\bcryptPrimitives.dll
(558.718): Break instruction exception - code 80000003 (first chance)
eax=00000000 ebx=00000000 ecx=859b0000 edx=00000000 esi=7ffde000 edi=00000000
eip=77bada8e esp=0018faec ebp=0018fb18 iopl=0         nv up ei pl zr na pe nc
cs=0023  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00000246
ntdll!LdrpDoDebuggerBreak+0x2b:
77bada8e cc              int     3
```

```
0:000> g
ModLoad: 76540000 76565000   C:\Windows\SysWOW64\IMM32.DLL
ModLoad: 76760000 76857000   C:\Windows\SysWOW64\MSCTF.dll
(558.3d4): Access violation - code c0000005 (first chance)
First chance exceptions are reported before any exception handling.
This exception may be expected and handled.
eax=07294fe8 ebx=00000003 ecx=07294fe8 edx=00000000 esi=ffffffff edi=071f0954
eip=6ff7b650 esp=0739fdc8 ebp=071dcf7c iopl=0         nv up ei pl nz na pe nc
cs=0023  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00010206
ACE_v6!ACE_Process_Options::creation_flags:
6ff7b650 8b4104          mov     eax,dword ptr [ecx+4] ds:002b:07294fec=????????
```

`ACE_v6!ACE_Process_Options::creation_flags`

According to:

http://www.dre.vanderbilt.edu/Doxygen/5.7.6/html/ace/a00507.html

`Set the creation flags to affect how a new process is spawned.`

How has `ecx` been affected earlier, looking backwards:

```
0:001> ub eip . L 16
ACE_v6!ACE_INET_Addr::is_ipv4_compat_ipv6+0x1d:
6ff7b62d cc              int     3
6ff7b62e cc              int     3
6ff7b62f cc              int     3
ACE_v6!ACE_Asynch_Result::ACE_Asynch_Result:
6ff7b630 8bc1            mov     eax,ecx
6ff7b632 8b4c2404        mov     ecx,dword ptr [esp+4]
6ff7b636 c70044e4ff6f    mov     dword ptr [eax],offset
ACE_v6!ACE_Asynch_Result::`vftable' (6fffe444)
6ff7b63c 894804          mov     dword ptr [eax+4],ecx
6ff7b63f c20400          ret     4
6ff7b642 cc              int     3
6ff7b643 cc              int     3
6ff7b644 cc              int     3
6ff7b645 cc              int     3
6ff7b646 cc              int     3
6ff7b647 cc              int     3
6ff7b648 cc              int     3
6ff7b649 cc              int     3
```

```
6ff7b64a cc              int     3
6ff7b64b cc              int     3
6ff7b64c cc              int     3
6ff7b64d cc              int     3
6ff7b64e cc              int     3
6ff7b64f cc              int     3
```

it takes the value at `[esp+4]`

Looking at the call stack and first three parameters:

```
0:001> kb
 # ChildEBP RetAddr      Args to Child
WARNING: Stack unwind information not available. Following frames may be wrong.
00 0739fdc4 6ffeb813     071dcf7c 071dcf7c 071dcf50
ACE_v6!ACE_Process_Options::creation_flags
01 0739fdec 6ffaefe8     071dced0 6ffed419 53e7b4d5
ACE_v6!ACE_WFMO_Reactor_Handler_Repository::make_changes_in_current_infos+0x163
02 0739fdf4 6ffed419     53e7b4d5 00000000 071dced0
ACE_v6!ACE_WFMO_Reactor_Handler_Repository::make_changes+0x8
03 0739fe20 6ffaf1d8     6ffaf1b2 53e7b489 00000000
ACE_v6!ACE_WFMO_Reactor::update_state+0x119
04 00000000 00000000     00000000 00000000 00000000
ACE_v6!ACE_WFMO_Reactor::safe_dispatch+0x88
```

For line 00, we see two times `071dcf7c` passed as argument to `Process Options...`

Looking at the heap:

```
0:001> !heap -p -a 071dcf7c
    address 071dcf7c found in
    _DPH_HEAP_ROOT @ 6b41000
    in busy allocation (  DPH_HEAP_BLOCK:         UserAddr         UserSize -
VirtAddr         VirtSize)
                                 73a0e38:          71dced0              130 -
71dc000             2000
          ACE_v6!ACE_WFMO_Reactor::`vftable'
    70088d9c verifier!AVrfDebugPageHeapAllocate+0x0000023c
    77bdfb79 ntdll!RtlDebugAllocateHeap+0x00000032
    77b79a73 ntdll!RtlpAllocateHeap+0x0003914a
    77b40b43 ntdll!RtlAllocateHeap+0x0000014c
```

```
    6fe93db8 MSVCR90!malloc+0x00000079
[f:\dd\vctools\crt_bld\self_x86\crt\src\malloc.c @ 163]
    6fe93eb8 MSVCR90!operator new+0x0000001f
[f:\dd\vctools\crt_bld\self_x86\crt\src\new.cpp @ 59]
    6fff4853 ACE_v6!std::_Init_locks::operator=+0x0000029d
    6ffcb0b7 ACE_v6!ACE_Reactor::ACE_Reactor+0x00000057
    6ffc1004 ACE_v6!ACE_OS_Thread_Adapter::invoke+0x00000074
    6fe53433 MSVCR90!_callthreadstartex+0x0000001b
[f:\dd\vctools\crt_bld\self_x86\crt\src\threadex.c @ 348]
    6fe534c7 MSVCR90!_threadstartex+0x00000069
[f:\dd\vctools\crt_bld\self_x86\crt\src\threadex.c @ 326]
    7575919f KERNEL32!BaseThreadInitThunk+0x0000000e
    77b4a8cb ntdll!__RtlUserThreadStart+0x00000020
    77b4a8a1 ntdll!_RtlUserThreadStart+0x0000001b
```

Looking at the return address of line 00:

```
0:001> ub 6ffeb813
ACE_v6!ACE_WFMO_Reactor_Handler_Repository::make_changes_in_current_infos+0x149:
6ffeb7f9 893481          mov     dword ptr [ecx+eax*4],esi
6ffeb7fc 017514          add     dword ptr [ebp+14h],esi
6ffeb7ff 8b4c2414        mov     ecx,dword ptr [esp+14h]
6ffeb803 3bca            cmp     ecx,edx
6ffeb805 7439            je
ACE_v6!ACE_WFMO_Reactor_Handler_Repository::make_changes_in_current_infos+0x190
(6ffeb840)
6ffeb807 e8b4dafaff      call
ACE_v6!ACE_Event_Handler::reference_counting_policy (6ff992c0)
6ffeb80c 8bc8            mov     ecx,eax
6ffeb80e e83dfef8ff      call    ACE_v6!ACE_Process_Options::creation_flags
(6ff7b650)
```

Still looking backwards, restart by setting `bp 6ffeb7fc`:

```
(ba4.470): Break instruction exception - code 80000003 (first chance)
eax=00000000 ebx=00000000 ecx=2ee70000 edx=00000000 esi=7ffde000 edi=00000000
eip=77bada8e esp=0018faec ebp=0018fb18 iopl=0         nv up ei pl zr na pe nc
cs=0023  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00000246
ntdll!LdrpDoDebuggerBreak+0x2b:
77bada8e cc              int     3
```

```
0:000> bp 6ffeb7fc
0:000> g
ModLoad: 76540000 76565000   C:\Windows\SysWOW64\IMM32.DLL
ModLoad: 76760000 76857000   C:\Windows\SysWOW64\MSCTF.dll
Breakpoint 0 hit
eax=00000003 ebx=00000003 ecx=0710ef00 edx=00000000 esi=ffffffff edi=07110954
eip=6ffeb7fc esp=072bfdcc ebp=070fcf7c iopl=0         nv up ei pl zr na pe nc
cs=0023  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00000246
ACE_v6!ACE_WFMO_Reactor_Handler_Repository::make_changes_in_current_infos+0x14c:
6ffeb7fc 017514          add     dword ptr [ebp+14h],esi
ss:002b:070fcf90=00000004

0:001> u
ACE_v6!ACE_WFMO_Reactor_Handler_Repository::make_changes_in_current_infos+0x14c:
6ffeb7fc 017514          add     dword ptr [ebp+14h],esi
6ffeb7ff 8b4c2414        mov     ecx,dword ptr [esp+14h]
6ffeb803 3bca            cmp     ecx,edx
6ffeb805 7439            je
ACE_v6!ACE_WFMO_Reactor_Handler_Repository::make_changes_in_current_infos+0x190
(6ffeb840)
6ffeb807 e8b4dafaff      call
ACE_v6!ACE_Event_Handler::reference_counting_policy (6ff992c0)
6ffeb80c 8bc8            mov     ecx,eax
6ffeb80e e83dfef8ff      call    ACE_v6!ACE_Process_Options::creation_flags
(6ff7b650)
6ffeb813 8b742414        mov     esi,dword ptr [esp+14h]
0:001> dc esi
ffffffff  ???????? ???????? ???????? ????????  ????????????????
0000000f  ???????? ???????? ???????? ????????  ????????????????
0000001f  ???????? ???????? ???????? ????????  ????????????????
0000002f  ???????? ???????? ???????? ????????  ????????????????
0000003f  ???????? ???????? ???????? ????????  ????????????????
0000004f  ???????? ???????? ???????? ????????  ????????????????
0000005f  ???????? ???????? ???????? ????????  ????????????????
0000006f  ???????? ???????? ???????? ????????  ????????????????
```

Keep going backwards:

From `kb` output:

```
02 0739fdf4 6ffed419     53e7b4d5 00000000 071dced0
ACE_v6!ACE_WFMO_Reactor_Handler_Repository::make_changes+0x8
```

Restart using `bp 6ffed419`:

```
(86c.224): Break instruction exception - code 80000003 (first chance)
eax=00000000 ebx=00000000 ecx=fdaf0000 edx=00000000 esi=7ffde000 edi=00000000
eip=77bada8e esp=0018faec ebp=0018fb18 iopl=0         nv up ei pl zr na pe nc
cs=0023  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00000246
ntdll!LdrpDoDebuggerBreak+0x2b:
77bada8e cc              int     3
0:000> bp 6ffed419
0:000> g
ModLoad: 76540000 76565000   C:\Windows\SysWOW64\IMM32.DLL
ModLoad: 76760000 76857000   C:\Windows\SysWOW64\MSCTF.dll
Breakpoint 0 hit
eax=00000000 ebx=0732cf50 ecx=ffffffff edx=c0c00001 esi=0732ced0 edi=0732cf7c
eip=6ffed419 esp=074efdfc ebp=00000000 iopl=0         nv up ei pl zr na pe nc
cs=0023  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00000246
ACE_v6!ACE_WFMO_Reactor::update_state+0x119:
6ffed419 8b07            mov     eax,dword ptr [edi]  ds:002b:0732cf7c=
{ACE_v6!ACE_WFMO_Reactor_Handler_Repository::`vftable' (70001ba8)}


0:001> u
ACE_v6!ACE_WFMO_Reactor::update_state+0x119:
6ffed419 8b07            mov     eax,dword ptr [edi]
6ffed41b 8b5004          mov     edx,dword ptr [eax+4]
6ffed41e 8bcf            mov     ecx,edi
6ffed420 ffd2            call    edx
6ffed422 84c0            test    al,al
6ffed424 75ea            jne     ACE_v6!ACE_WFMO_Reactor::update_state+0x110
(6ffed410)
6ffed426 8b06            mov     eax,dword ptr [esi]
6ffed428 8b9050010000    mov     edx,dword ptr [eax+150h]


0:001> t
eax=70001ba8 ebx=0732cf50 ecx=ffffffff edx=c0c00001 esi=0732ced0 edi=0732cf7c
```

```
eip=6ffed41b esp=074efdfc ebp=00000000 iopl=0         nv up ei pl zr na pe nc
cs=0023  ss=002b  ds=002b  es=002b  fs=0053  gs=002b             efl=00000246
ACE_v6!ACE_WFMO_Reactor::update_state+0x11b:
6ffed41b 8b5004          mov     edx,dword ptr [eax+4] ds:002b:70001bac=
{ACE_v6!ACE_WFMO_Reactor_Handler_Repository::changes_required (6ffaefb0)}
```

Looking into `eax`:

```
0:001> dps 70001ba8
70001ba8  6ffaee20
ACE_v6!ACE_WFMO_Reactor_Handler_Repository::ACE_WFMO_Reactor_Handler_Repository+
0x60
70001bac  6ffaefb0 ACE_v6!ACE_WFMO_Reactor_Handler_Repository::changes_required
70001bb0  6ffaefe0 ACE_v6!ACE_WFMO_Reactor_Handler_Repository::make_changes
70001bb4  00000000
70001bb8  70009b88 ACE_v6!ACE_WIN32_Proactor::`vftable'+0x3364
70001bbc  6ffaf690 ACE_v6!ACE_Msg_WFMO_Reactor::`default constructor
closure'+0x20
70001bc0  6ffedc90 ACE_v6!ACE_WFMO_Reactor::open
70001bc4  6ffebbb0 ACE_v6!ACE_Event_Handler::handle_timeout
70001bc8  6ffebbc0 ACE_v6!ACE_WFMO_Reactor::set_sig_handler
70001bcc  6ffdd420 ACE_v6!ACE_Dynamic_Message_Strategy::dynamic_priority_max
70001bd0  6ffebbf0 ACE_v6!ACE_WFMO_Reactor::timer_queue
70001bd4  6ffed880 ACE_v6!ACE_WFMO_Reactor::close
70001bd8  6ffebb90 ACE_v6!ACE_Recursive_Thread_Mutex::acquire
70001bdc  6ffaf3b0 ACE_v6!ACE_WFMO_Reactor::handle_events
70001be0  6ffaf3b0 ACE_v6!ACE_WFMO_Reactor::handle_events
70001be4  6ffaf3d0 ACE_v6!ACE_Msg_WFMO_Reactor::alertable_handle_events
70001be8  6ffaf3d0 ACE_v6!ACE_Msg_WFMO_Reactor::alertable_handle_events
70001bec  6ff71b90 ACE_v6!ACE_WFMO_Reactor::deactivated
70001bf0  6ffaf0f0 ACE_v6!ACE_WFMO_Reactor::deactivate
70001bf4  6ffaf210 ACE_v6!ACE_WFMO_Reactor::register_handler
70001bf8  6ffaf200 ACE_v6!ACE_WFMO_Reactor::register_handler
70001bfc  6ffafa90 ACE_v6!ACE_WFMO_Reactor::register_handler
70001c00  6ffaf9e0 ACE_v6!ACE_WFMO_Reactor::register_handler
70001c04  6ffaf7d0 ACE_v6!ACE_WFMO_Reactor::register_handler
70001c08  6ffaf930 ACE_v6!ACE_WFMO_Reactor::register_handler
70001c0c  6ffaf880 ACE_v6!ACE_WFMO_Reactor::register_handler
70001c10  6ffaf2a0 ACE_v6!ACE_WFMO_Reactor::remove_handler
70001c14  6ff7e390 ACE_v6!ACE_WFMO_Reactor::remove_handler
70001c18  6ffafe70 ACE_v6!ACE_WFMO_Reactor::remove_handler
```

```
70001c1c  6ffafe60 ACE_v6!ACE_WFMO_Reactor::remove_handler
70001c20  6ffafe30 ACE_v6!ACE_WFMO_Reactor::remove_handler
70001c24  6ffb0020 ACE_v6!ACE_WFMO_Reactor::suspend_handler
```

We see seven call to: `ACE_v6!ACE_WFMO_Reactor::register_handler`
and five to: `ACE_v6!ACE_WFMO_Reactor::remove_handler`

```
0:001> dps esi
0732ced0  7000603c ACE_v6!ACE_WFMO_Reactor::`vftable'
0732ced4  0733cff8
0732ced8  c0c0c001
0732cedc  0732ef78
0732cee0  c0c00101
0732cee4  07346f30
0732cee8  c0c0c001
0732ceec  32333730
0732cef0  43454543
0732cef4  36353132
0732cef8  c0c0c000
0732cefc  c0c0c0c0
0732cf00  c0c0c0c0
0732cf04  c0c0c0c0
0732cf08  c0c0c0c0
0732cf0c  c0c0c0c0
0732cf10  c0c0c0c0
0732cf14  c0c0c0c0
0732cf18  c0c0c0c0
0732cf1c  c0c0c0c0
0732cf20  c0c0c0c0
0732cf24  c0c0c0c0
0732cf28  c0c0c0c0
0732cf2c  c0c0c0c0
0732cf30  c0c0c0c0
0732cf34  c0c0c0c0
0732cf38  c0c0c0c0
0732cf3c  c0c0c0c0
0732cf40  c0c0c0c0
0732cf44  c0c0c0c0
0732cf48  c0c0c0c0
0732cf4c  c0c0c0c0
```

```
0:001> dps edi
0732cf7c  70001ba8 ACE_v6!ACE_WFMO_Reactor_Handler_Repository::`vftable'
0732cf80  0732ced0
0732cf84  00000040
0732cf88  0733ef00
0732cf8c  07340900
0732cf90  00000003
0732cf94  07344800
0732cf98  00000000
0732cf9c  07342800
0732cfa0  00000000
0732cfa4  00000000
0732cfa8  00000000
0732cfac  00000000
0732cfb0  00000134
0732cfb4  c0c0c000
0732cfb8  00000138
0732cfbc  c0c0c000
0732cfc0  70001b54 ACE_v6!ACE_Wakeup_All_Threads_Handler::`vftable'
0732cfc4  00000001
0732cfc8  00000000
0732cfcc  00000000
0732cfd0  6fffe1ec
ACE_v6!ACE_Event_Handler::Reference_Counting_Policy::`vftable'
0732cfd4  00000001
0732cfd8  0000013c
0732cfdc  c0c0c000
0732cfe0  00000000
0732cfe4  000009b4
0732cfe8  00000000
0732cfec  000009b4
0732cff0  00000130
0732cff4  00000134
0732cff8  c0c0c001
```

`0xc0c0c0c0` is a marker for uninitialized data.

```
0:001> !heap -p -a ecx
    address 0732cf7c found in
    _DPH_HEAP_ROOT @ 6c51000
```

```
    in busy allocation (  DPH_HEAP_BLOCK:         UserAddr         UserSize -
VirtAddr         VirtSize)
                                 74f0e38:          732ced0              130 -
732c000             2000
          ACE_v6!ACE_WFMO_Reactor::`vftable'
    70088d9c verifier!AVrfDebugPageHeapAllocate+0x0000023c
    77bdfb79 ntdll!RtlDebugAllocateHeap+0x00000032
    77b79a73 ntdll!RtlpAllocateHeap+0x0003914a
    77b40b43 ntdll!RtlAllocateHeap+0x0000014c
    6fe93db8 MSVCR90!malloc+0x00000079
[f:\dd\vctools\crt_bld\self_x86\crt\src\malloc.c @ 163]
    6fe93eb8 MSVCR90!operator new+0x0000001f
[f:\dd\vctools\crt_bld\self_x86\crt\src\new.cpp @ 59]
    6fff4853 ACE_v6!std::_Init_locks::operator=+0x0000029d
    6ffcb0b7 ACE_v6!ACE_Reactor::ACE_Reactor+0x00000057
    6ffc1004 ACE_v6!ACE_OS_Thread_Adapter::invoke+0x00000074
    6fe53433 MSVCR90!_callthreadstartex+0x0000001b
[f:\dd\vctools\crt_bld\self_x86\crt\src\threadex.c @ 348]
    6fe534c7 MSVCR90!_threadstartex+0x00000069
[f:\dd\vctools\crt_bld\self_x86\crt\src\threadex.c @ 326]
    7575919f KERNEL32!BaseThreadInitThunk+0x0000000e
    77b4a8cb ntdll!__RtlUserThreadStart+0x00000020
    77b4a8a1 ntdll!_RtlUserThreadStart+0x0000001b
```

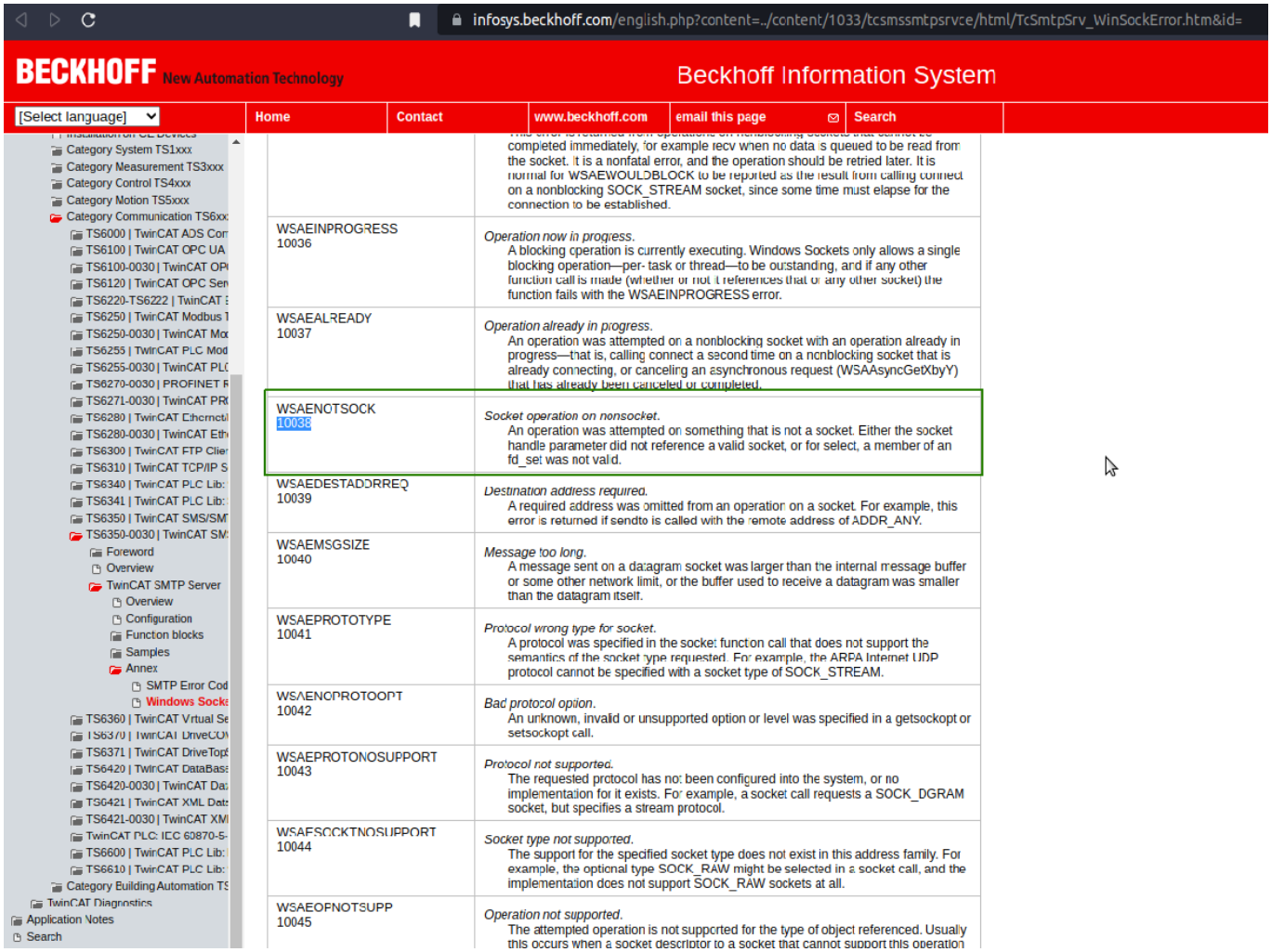
It's difficult to keep track of the "callings" by rewinding like this.

Exploring through other debugging features of windbg:

Getting kernel32!GetLastError after AV

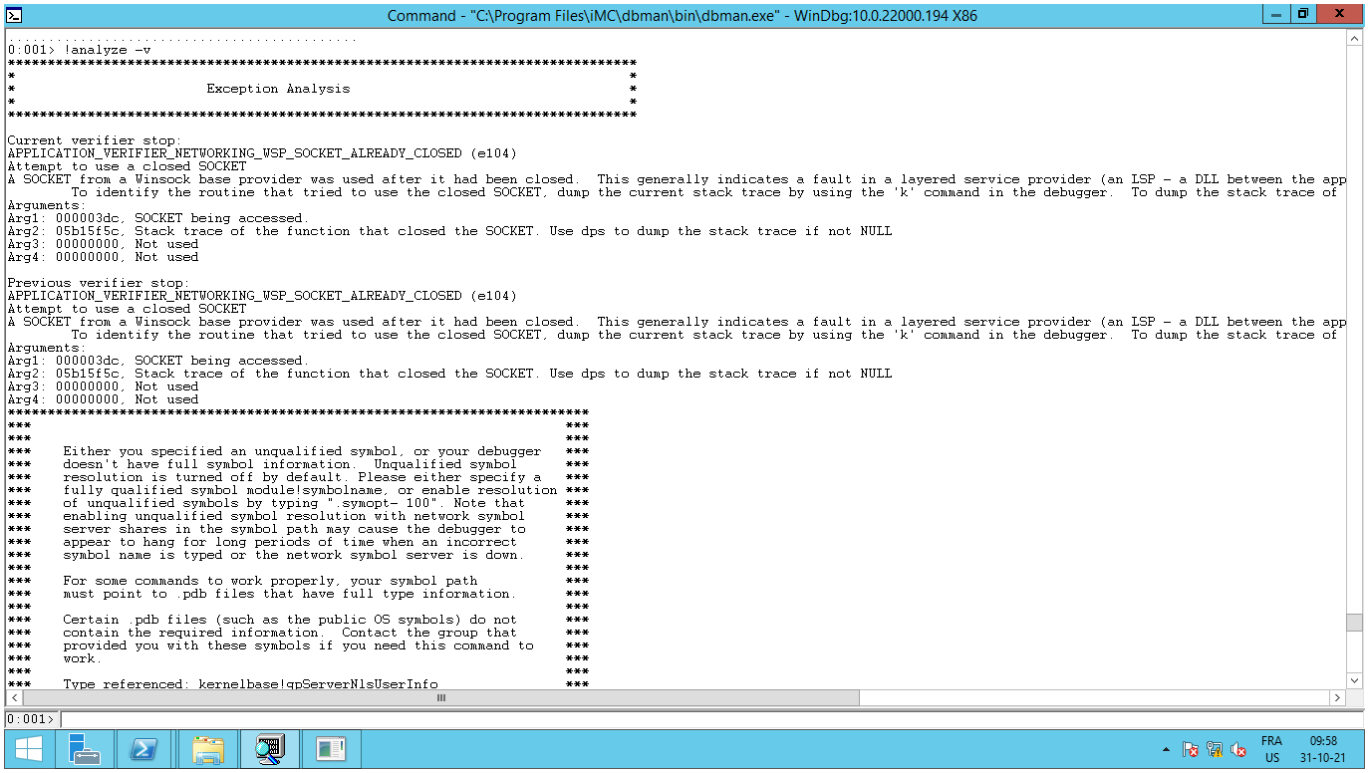
```
0:001> !gle
LastErrorValue: (WinSock) 0x2736 (10038) - An operation was attempted on
something that is not a socket.
LastStatusValue: (NTSTATUS) 0xc0000008 - An invalid HANDLE was specified
```

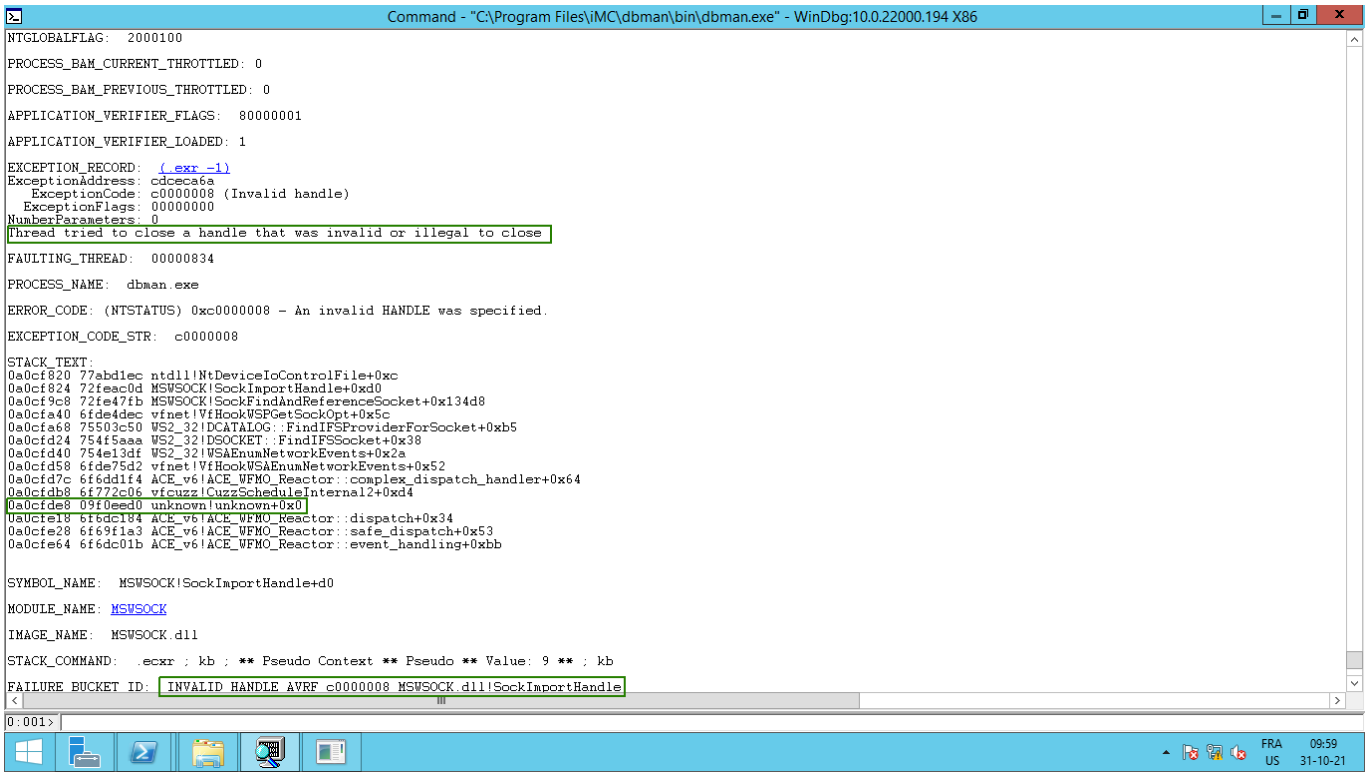
A bit more detail on the error code:

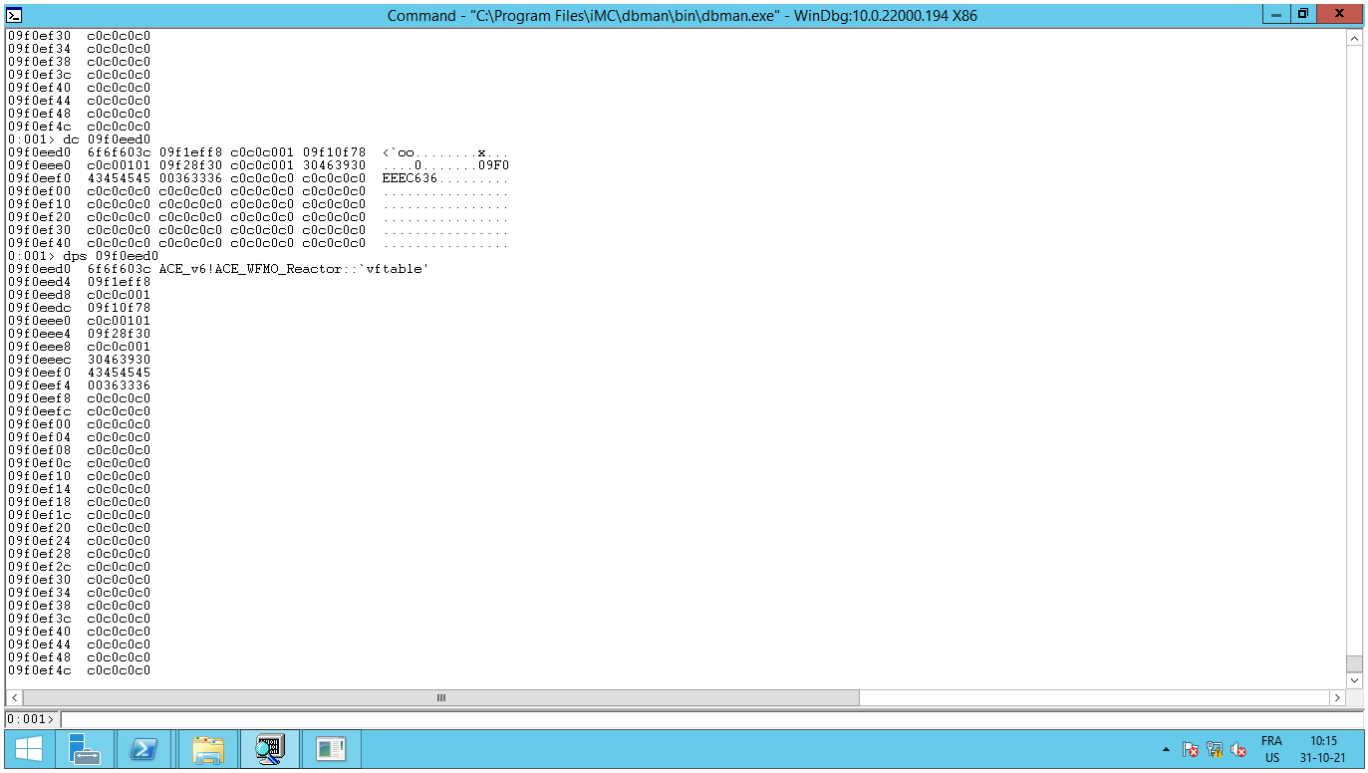


https://docs.microsoft.com/en-us/windows/win32/api/errhandlingapi/nf-errhandlingapi-getlasterror

Follow up this exception/handle issue in windbg:


Command - "C:\Program Files\iMC\dbman\bin\dbman.exe" - WinDbg:10.0.22000.194 X86
0:001> !analyze -v
Exception Analysis
Current verifier stop:
APPLICATION_VERIFIER_NETWORKING_WSP_SOCKET_ALREADY_CLOSED (e104)
Attempt to use a closed SOCKET
A SOCKET from a Winsock base provider was used after it had been closed. This generally indicates a fault in a layered service provider (an LSP - a DLL between the app
To identify the routine that tried to use the closed SOCKET, dump the current stack trace by using the 'k' command in the debugger. To dump the stack trace of
Arguments:
Arg1: 000003dc, SOCKET being accessed.
Arg2: 05b15f5c, Stack trace of the function that closed the SOCKET. Use dps to dump the stack trace if not NULL
Arg3: 00000000, Not used
Arg4: 00000000, Not used
Previous verifier stop:
APPLICATION_VERIFIER_NETWORKING_WSP_SOCKET_ALREADY_CLOSED (e104)
Attempt to use a closed SOCKET
A SOCKET from a Winsock base provider was used after it had been closed. This generally indicates a fault in a layered service provider (an LSP - a DLL between the app
To identify the routine that tried to use the closed SOCKET, dump the current stack trace by using the 'k' command in the debugger. To dump the stack trace of
Arguments:
Arg1: 000003dc, SOCKET being accessed.
Arg2: 05b15f5c, Stack trace of the function that closed the SOCKET. Use dps to dump the stack trace if not NULL
Arg3: 00000000, Not used
Arg4: 00000000, Not used
*** Either you specified an unqualified symbol, or your debugger
*** doesn't have full symbol information. Unqualified symbol
*** resolution is turned off by default. Please either specify a
*** fully qualified symbol module!symbolname, or enable resolution
*** of unqualified symbols by typing ".symopt- 100". Note that
*** enabling unqualified symbol resolution with network symbol
*** server shares in the symbol path may cause the debugger to
*** appear to hang for long periods of time when an incorrect
*** symbol name is typed or the network symbol server is down.
*** For some commands to work properly, your symbol path
*** must point to .pdb files that have full type information.
*** Certain .pdb files (such as the public OS symbols) do not
*** contain the required information. Contact the group that
*** provided you with these symbols if you need this command to
*** work.
*** Type referenced: kernelbase!gpServerNlsUserInfo
0:001>
FRA
US
09:58
31-10-21


Command - "C:\Program Files\iMC\dbman\bin\dbman.exe" - WinDbg:10.0.22000.194 X86
NTGLOBALFLAG: 2000100
PROCESS_BAM_CURRENT_THROTTLED: 0
PROCESS_BAM_PREVIOUS_THROTTLED: 0
APPLICATION_VERIFIER_FLAGS: 80000001
APPLICATION_VERIFIER_LOADED: 1
EXCEPTION_RECORD: (.exr -1)
ExceptionAddress: cdceca6a
ExceptionCode: c0000008 (Invalid handle)
ExceptionFlags: 00000000
NumberParameters: 0
Thread tried to close a handle that was invalid or illegal to close
FAULTING_THREAD: 00000834
PROCESS_NAME: dbman.exe
ERROR_CODE: (NTSTATUS) 0xc0000008 - An invalid HANDLE was specified.
EXCEPTION_CODE_STR: c0000008
STACK_TEXT:
0a0cf820 77abd1ec ntdll!NtDeviceIoControlFile+0xc
0a0cf824 72feac0d MSWSOCK!SockImportHandle+0xd0
0a0cf9c8 72fe47fb MSWSOCK!SockFindAndReferenceSocket+0x134d8
0a0cfa40 6fde4dec vfnet!VfHookWSPGetSockOpt+0x5c
0a0cfa68 75503c50 WS2_32!DCATALOG::FindIFSProviderForSocket+0xb5
0a0cfd24 754f5aaa WS2_32!DSOCKET::FindIFSSocket+0x38
0a0cfd40 754e13df WS2_32!WSAEnumNetworkEvents+0x2a
0a0cfd58 6fde75d2 vfnet!VfHookWSAEnumNetworkEvents+0x52
0a0cfd7c 6f6dd1f4 ACE_v6!ACE_WFMO_Reactor::complex_dispatch_handler+0x64
0a0cfdb8 6f772c06 vfcuzz!CuzzScheduleInterna12+0xd4
0a0cfde8 09f0eed0 unknown!unknown+0xd0
0a0cfe18 6f6dc184 ACE_v6!ACE_WFMO_Reactor::dispatch+0x34
0a0cfe28 6f69f1a3 ACE_v6!ACE_WFMO_Reactor::safe_dispatch+0x53
0a0cfe64 6f6dc01b ACE_v6!ACE_WFMO_Reactor::event_handling+0xbb
SYMBOL_NAME: MSWSOCK!SockImportHandle+d0
MODULE_NAME: MSWSOCK
IMAGE_NAME: MSWSOCK.dll
STACK_COMMAND: .ecxr ; kb ; ** Pseudo Context ** Pseudo ** Value: 9 ** ; kb
FAILURE_BUCKET_ID: INVALID_HANDLE_AVRF_c0000008_MSWSOCK.dll!SockImportHandle
0:001>
FRA
US
09:59
31-10-21



Wrap up -- situation:

Exploitability:

As mentioned earlier, difficult it is to spot the vulnerability from a static analysis only perspective.

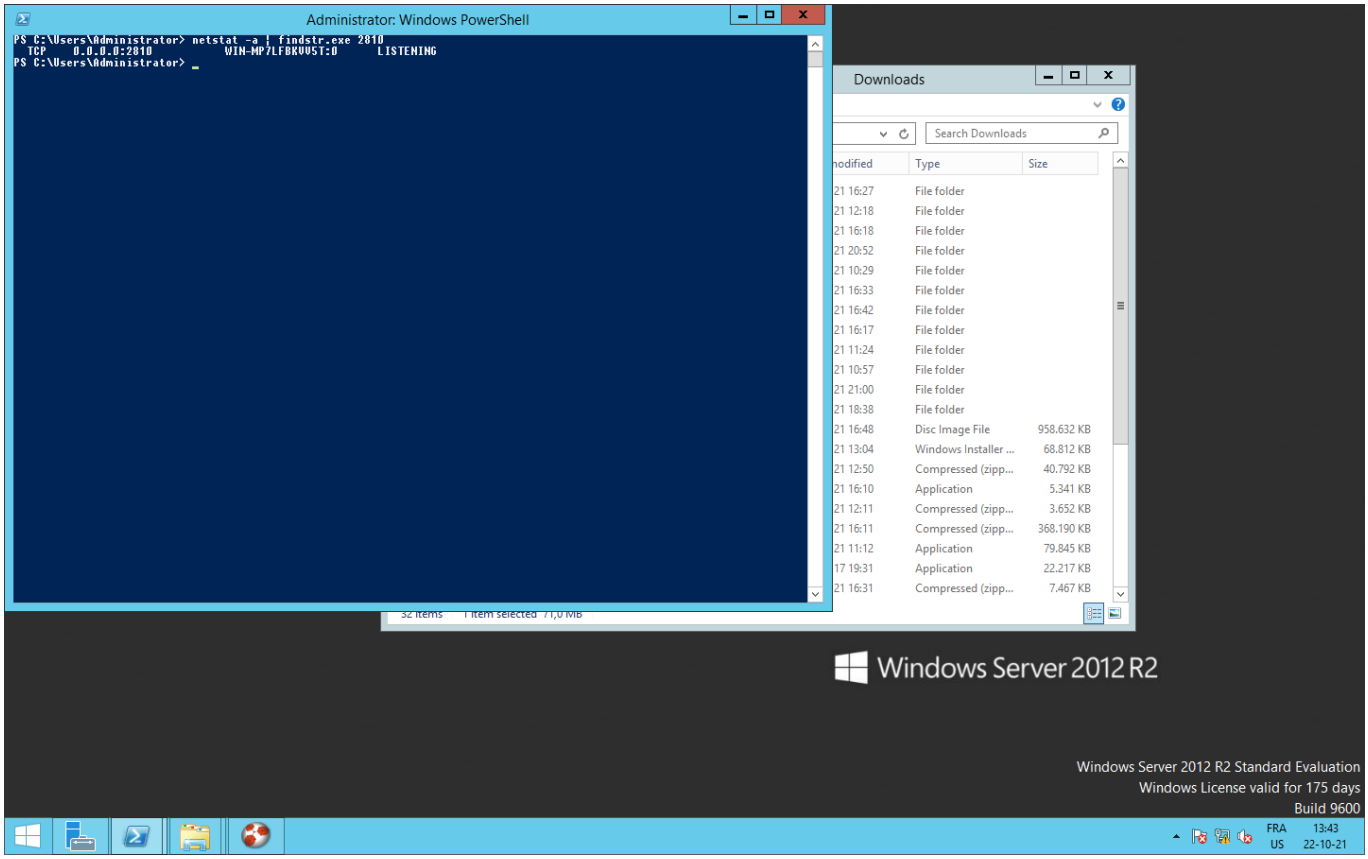
Stack-based buffer overflow

this section hasn't been thoroughly explored

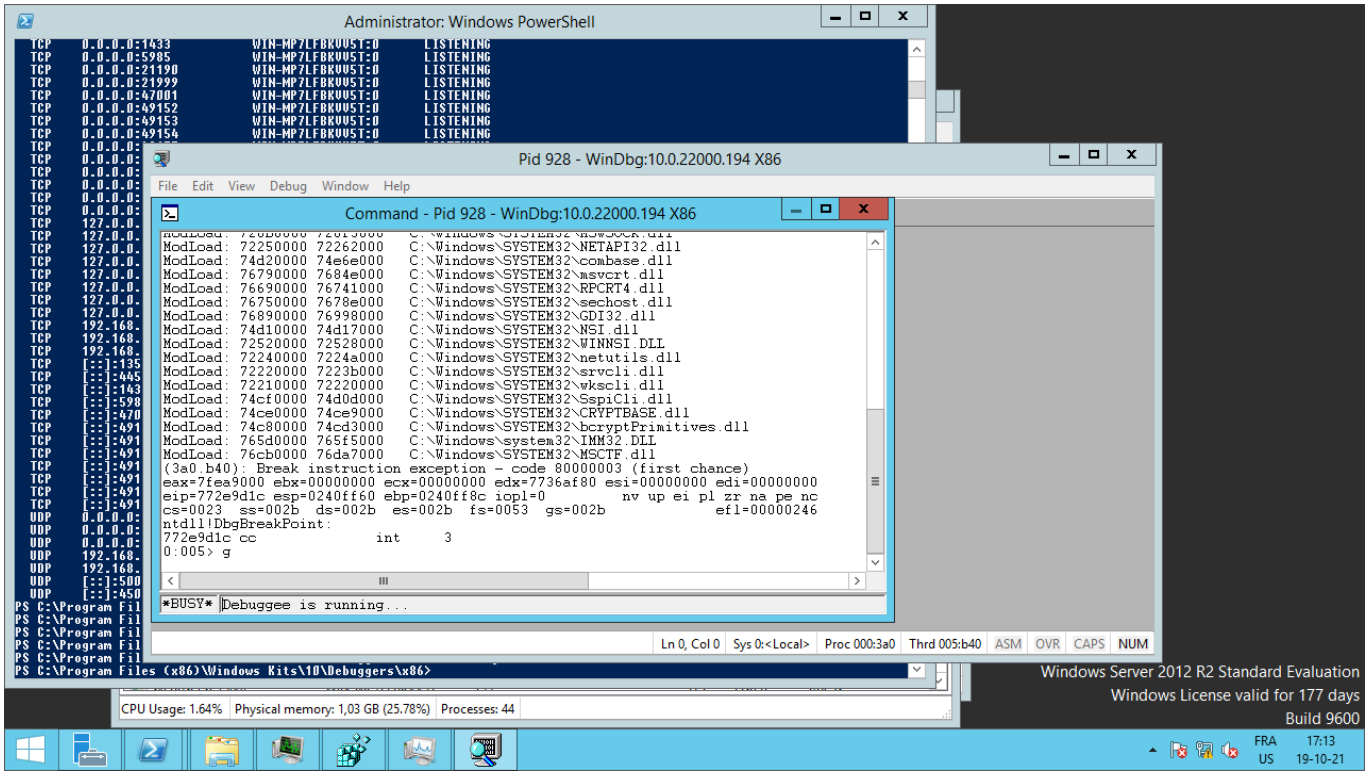
(iMC_PLAT_7.3_E0504_Ent_Win-003)

The following bug looks pretty much as a stack-based buffer overflow. At first look, it concerns the command opcode 10001.

To observe the behaviour of *dbman.exe* attaching a debugger can be handy. In a first place, I am going for winDbg and msecdbg extension. First, we assure that *dbman.exe* is running i.e. listening on port 2810:



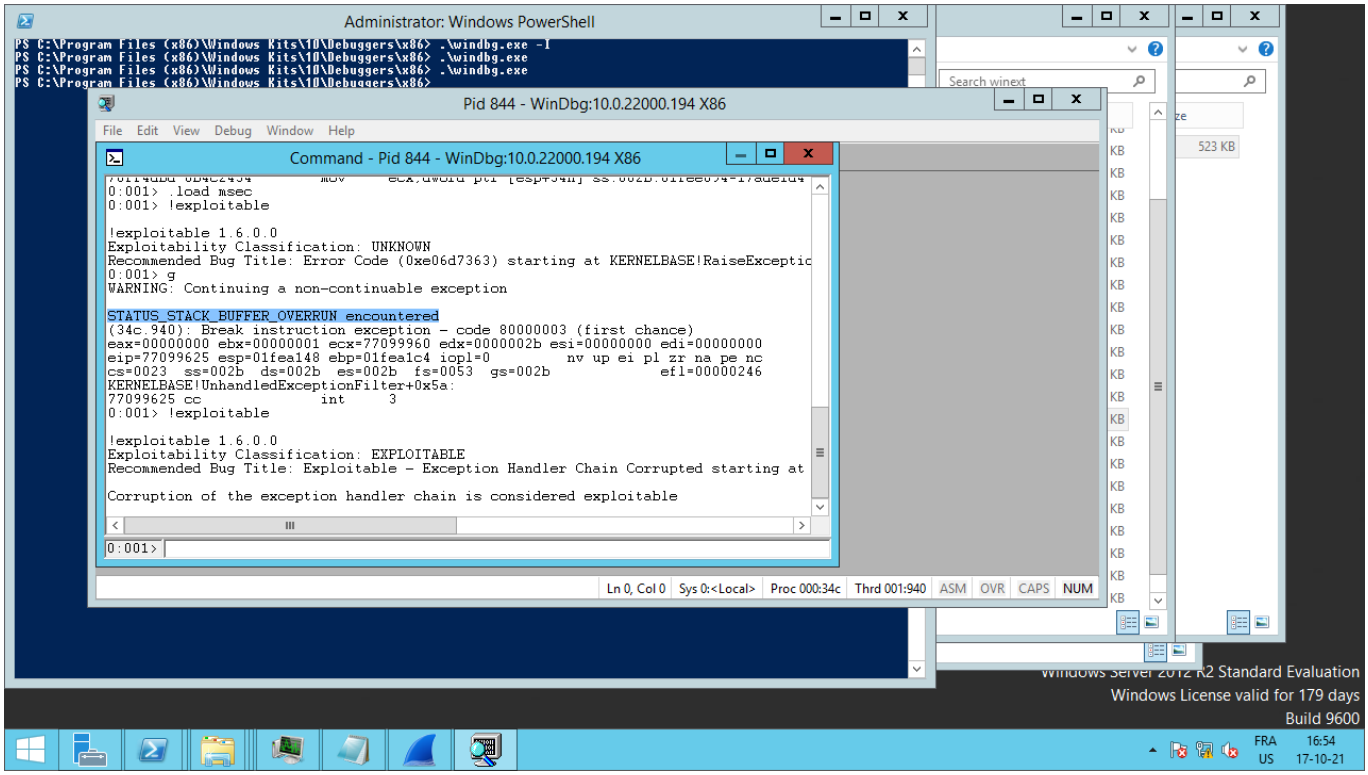
Attach WinDbg to dbman.exe:



Send Payload to port 2810:

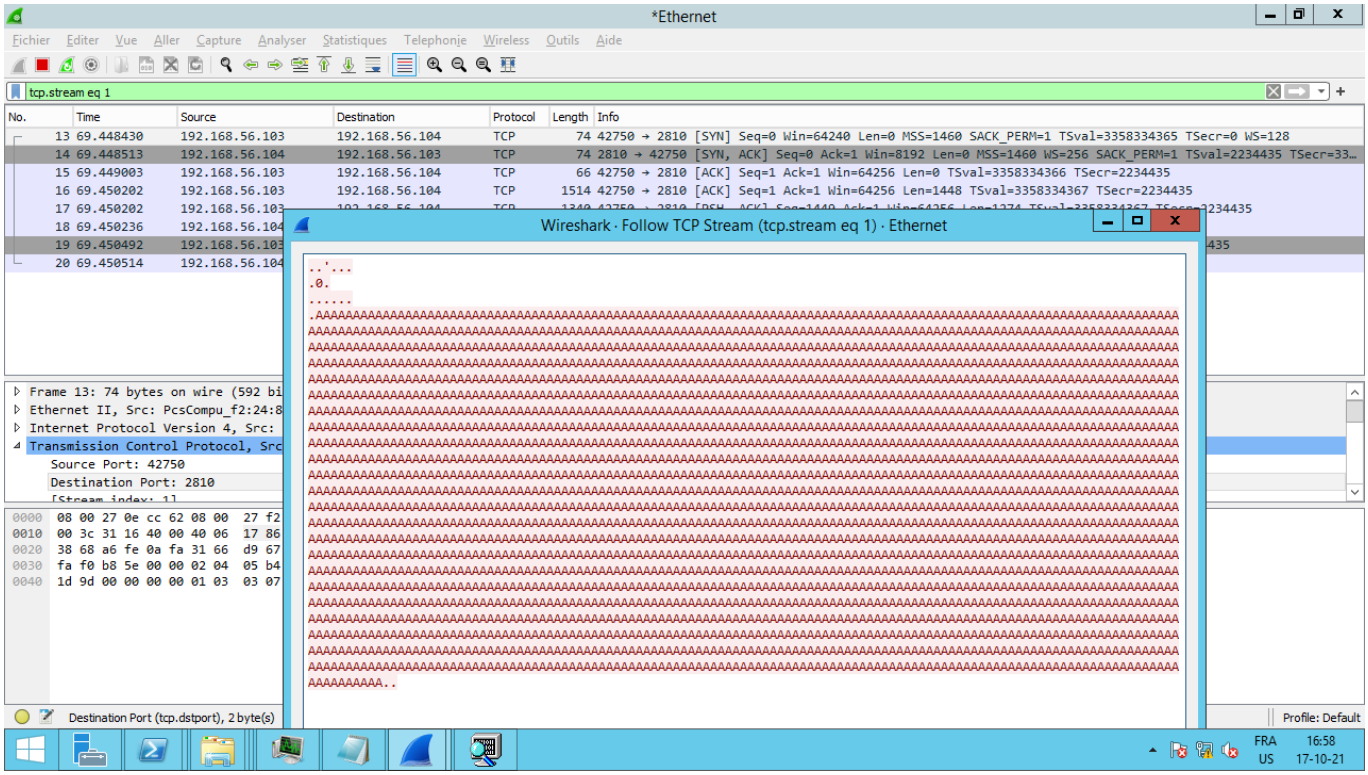
```
└─$ python3 fuzz.py
[+] Opening connection to 192.168.56.104 on port 2810: Done
[*] Closed connection to 192.168.56.104 port 2810
```

Observe behaviour, load msec extension and check exploitability:



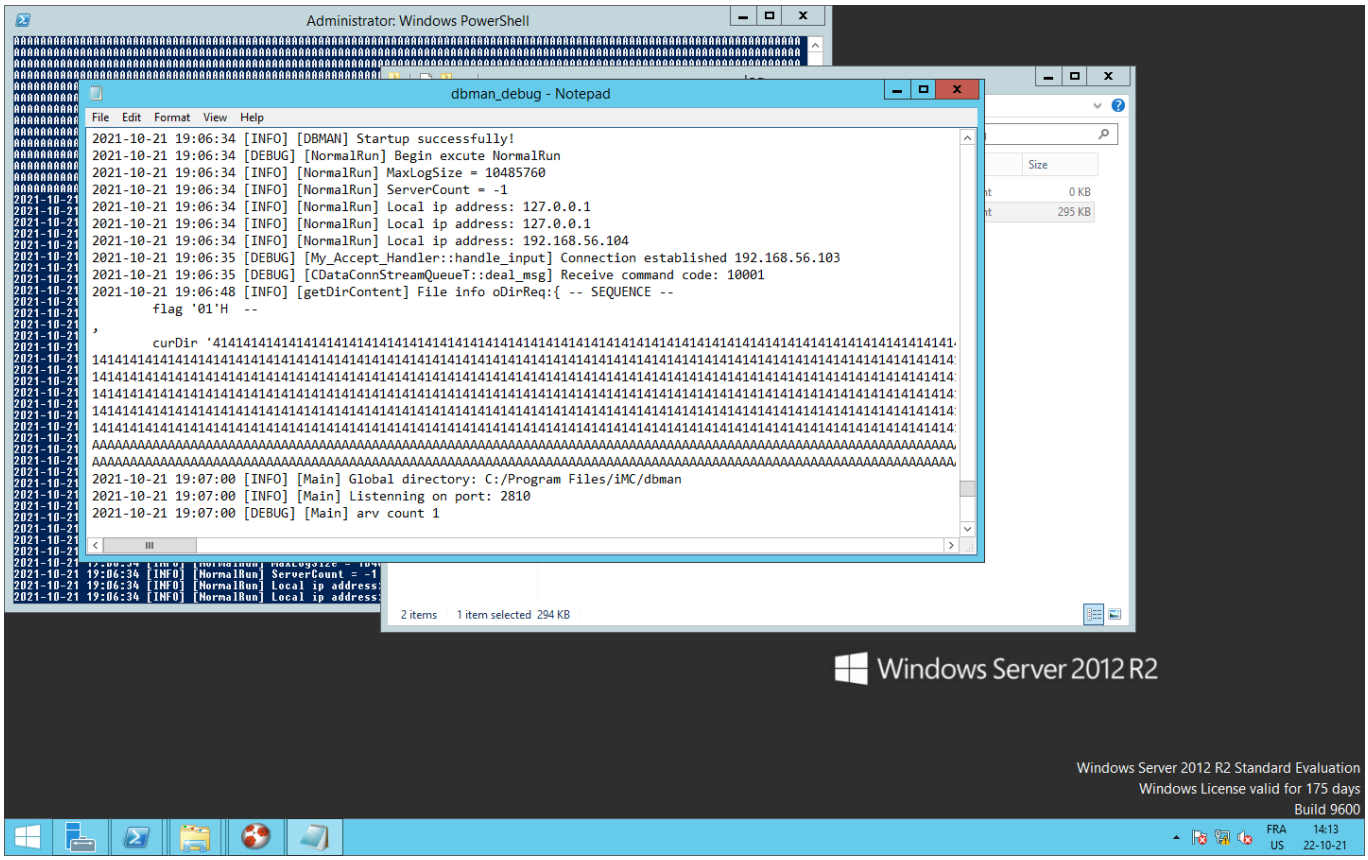
The msec extension tells us that it's potentially exploitable.

Looking at payload in wireshark:



Looking at the dbman_debug log file:

```
C:\Program Files\iMC\dbman\log\dbman_debug.log
```



Analyzing potential exploitability:

The exploitability requires at least the following two conditions:

- Can we "control" the instruction pointer?
- Can we store, and have enough space to place our payload/shellcode?

The question of PE file security mitigations comes into play. As previously mentioned, it seems that the *dbman.exe* has been compiled with /GS flag i.e., stack cookie.
A few possibilities exist to bypass this mitigation, one option would be to take advantage of the SEH records to avoid this protection.

Wrap up, backtracking and lessons learned so far

In this case, working backwards is a rather cumbersome process. Manually keeping track of the unfolding processes isn't straightforward.

In the case of UAF, the issue seems to be related to some handlers but couldn't pinpoint the location. Moreover, is this a particular case of a UAF e.g., a double free or other flavor cannot be determined with precision with this RE approach.

The approach taken was/is sub-optimal to say the least. The size of the software at rest should have acted as a warning.

This calls for better Free/Open Source software/tooling(or money), something like `strace` on steriods. Timeless debugging could probably have helped, as briefly discussed hereafter couldn't make use of windbg preview time-travel feature. To my knowledge and personal look up, the options in this field are rather sparse.

Haven't looked yet into this deeply but maybe it could be possible to lower the RE overhead using `panda-re` ; could it act as a timeless debugger.

Encountered Issues

Time-network link related:

- Dealing with large file size took me some time to download(network bandwidth quite low on my side, 4G modem/router)
- setting up the environments, downloading iso/vms.

Windows ISO deployment/installation issue

By using the following ISO:

https://archive.org/details/Windows_Server_2008_R2_x64.iso_reupload

I wasn't able to install stuff because of a missing service in MS Windows. After several attempts, I decided to try another version i.e. WinServer2012R2.

Database related issue

iMC accepts databases from Oracle, SQLServer and MySQL.

I couldn't make it work using different versions of MySQL on the same host(iMC and db on the same machine). Another possibility could have been to use a remote db on a RedHat/CentOS machine.

GitHub - Justinfact/imc-ilms: HPE IMC Linux MySQL Script (ILMS)

OpenVas 21.4.2

After HPE iMC installation, I wanted to quickly check using another logic if I could quickly confirm the vulnerability ; wasn't sucessful.

Tried unsuccessfully(obviously) running("manually") the following script from Tenable:

https://vulners.com/nessus/HP_INTELLIGENT_MANAGEMENT_CENTER_7_3_E0504P04.NASL

Windbg Preview

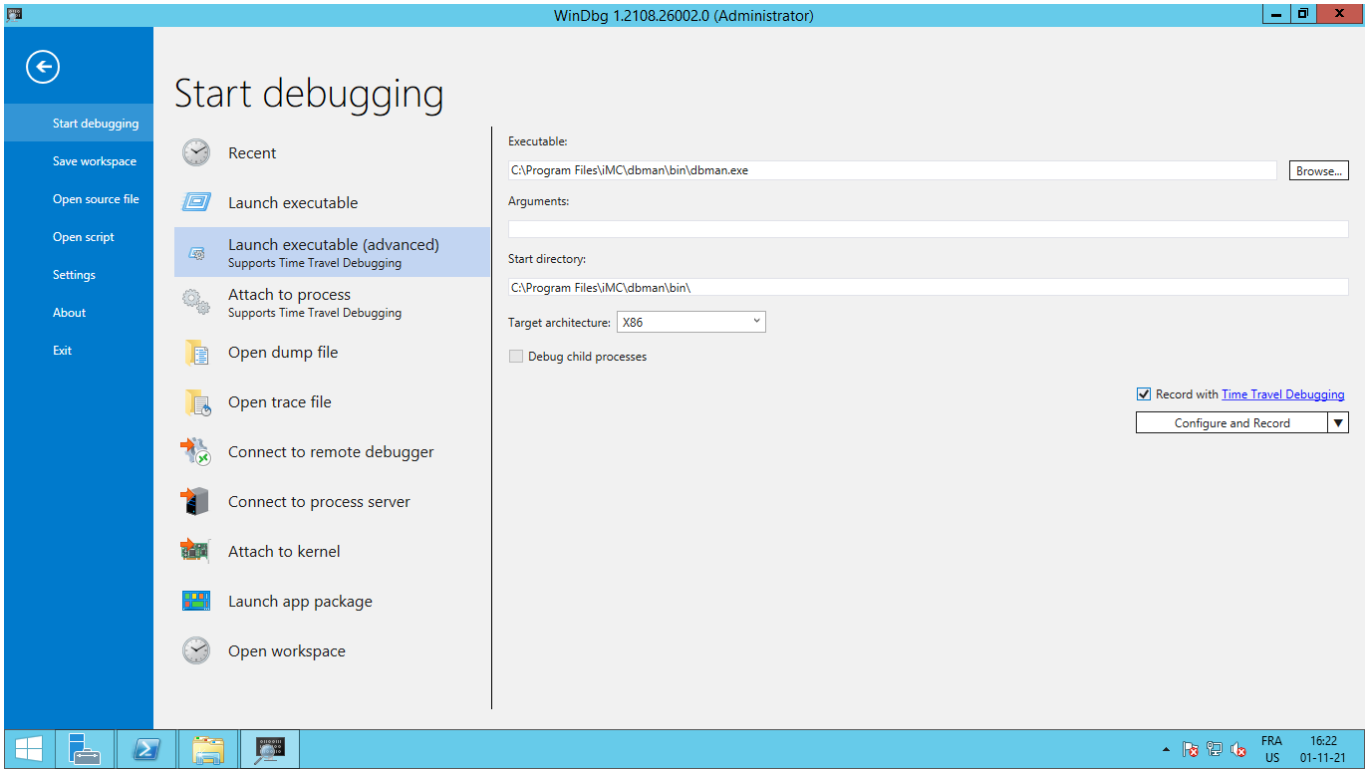
Managed to install windbg preview on MS WindowsServer2012R2 following:

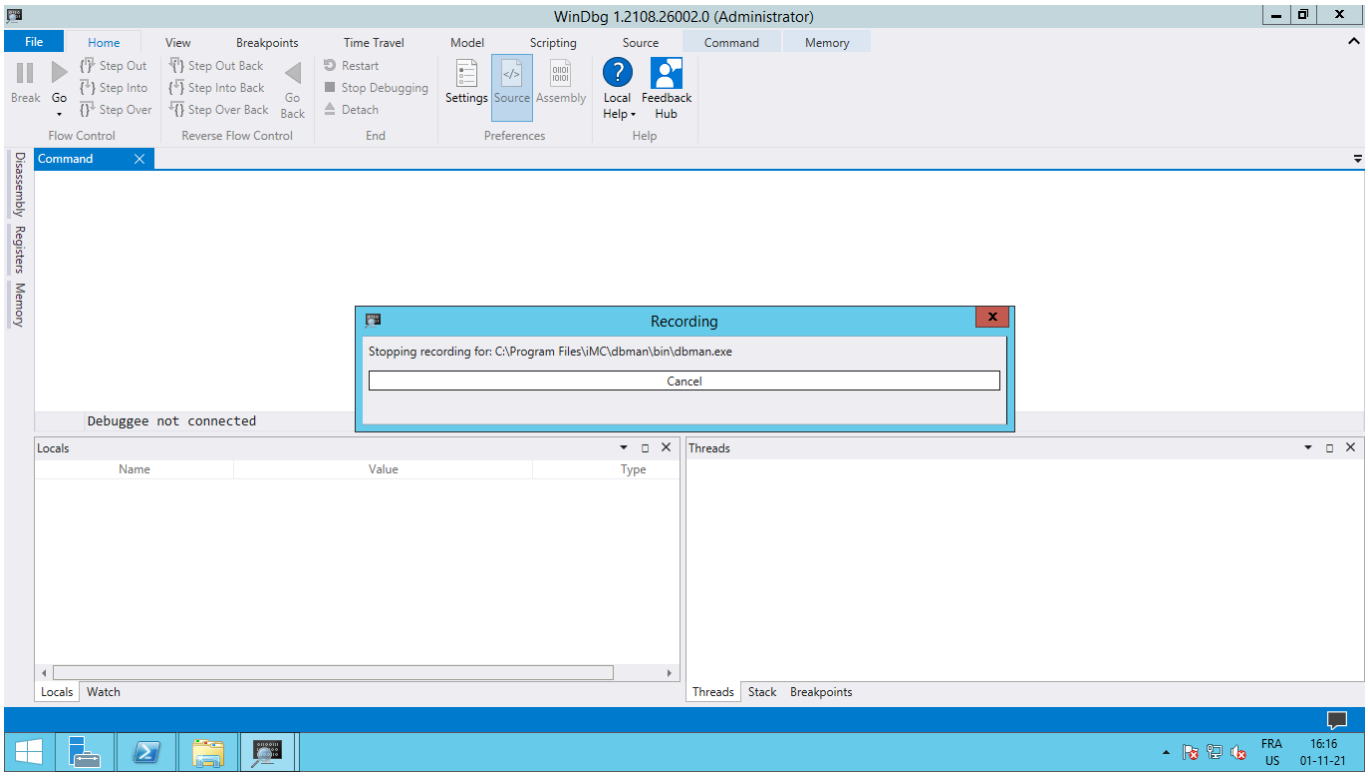
https://stackoverflow.com/questions/67971698/time-travel-debugging-how-to-install-it-without-the-store

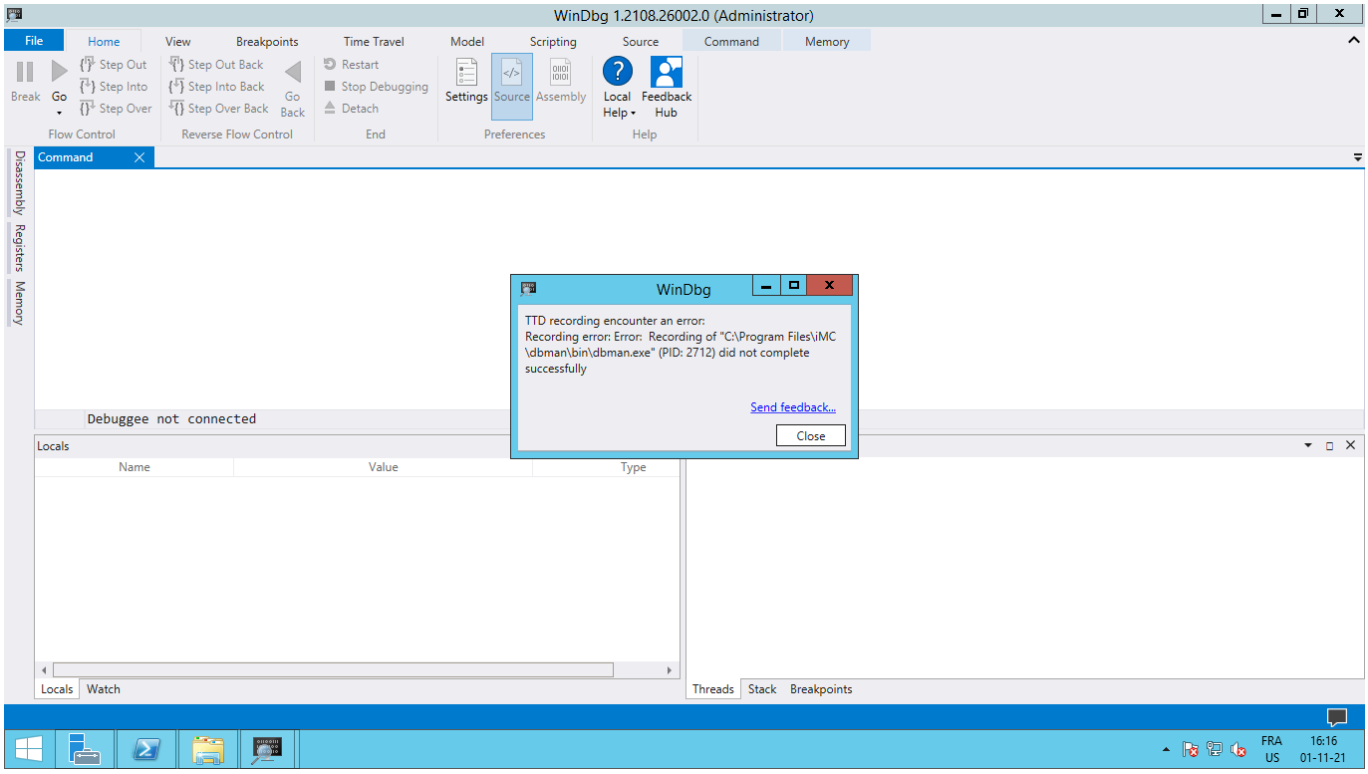
It comes down to downloading 7zip and the `.appx` file followed by:

- unarchiving the .appx using 7zip,
- launch `DbgX.Shell.exe` from within the extracted folder

For some reason, the recording of the time-travel for `dbman.exe` fails:


WinDbg 1.2108.26002.0 (Administrator)
Start debugging
Save workspace
Open source file
Open script
Settings
About
Exit
Start debugging
Recent
Launch executable
Launch executable (advanced)
Supports Time Travel Debugging
Attach to process
Supports Time Travel Debugging
Open dump file
Open trace file
Connect to remote debugger
Connect to process server
Attach to kernel
Launch app package
Open workspace
Executable:
C:\Program Files\iMC\dbman\bin\dbman.exe
Browse...
Arguments:
Start directory:
C:\Program Files\iMC\dbman\bin\
Target architecture:
X86
Debug child processes
Record with Time Travel Debugging
Configure and Record
FRA
US
16:22
01-11-21


WinDbg 1.2108.26002.0 (Administrator)
File
Home
View
Breakpoints
Time Travel
Model
Scripting
Source
Command
Memory
Break
Go
Step Out
Step Into
Step Over
Flow Control
Step Out Back
Step Into Back
Step Over Back
Go Back
Reverse Flow Control
Restart
Stop Debugging
Detach
End
Settings
Source
Assembly
Preferences
Local Help
Feedback Hub
Help
Command
Disassembly
Registers
Memory
Recording
Stopping recording for: C:\Program Files\iMC\dbman\bin\dbman.exe
Cancel
Debuggee not connected
Locals
Name
Value
Type
Threads
Locals
Watch
Threads
Stack
Breakpoints
FRA
US
16:16
01-11-21



Using that same feature on a simpler process e.g., a *ping* issued from a `cmd.exe` works.

At this point, the cause of failure hasn't been investigated, maybe it's linked to the daemon/service nature of dbman executable.

Links:
https://docs.microsoft.com/en-us/windows-hardware/drivers/debugger/time-travel-debugging-object-model

https://www.microsoft.com/en-us/p/windbg-preview/9pgjgd53tn86

iMC general deployment notes

Log:

Tried deploying on:

- Win7sp1 (AMD64) - vm
- Win10 (AMD64) - physical host
- Win2k8 (x86) - vm
- Win2kR2 (AMD64) - vm
- Win2k12R2 (AMD64) - vm

using variations for SQLServer versions.

Looking for other tooling

The motivation for accessing iMC under panda-re is because we want to be able to go back in time in the execution flow. As previously described, on a `WinServer2012R2`, it is possible to deploy windbg preview with the time-travel feature.
I couldn't make it work/apply it with `dbman.exe`.

One option that could help is to check if panda-re may be of some assistance.
Theoretically, this looks like a really powerful tool.

PANDA: Platform for Architecture Neutral Dynamic Analysis

Deploying Intelligent Management Center PLAT 7.3 (E0504P04) to a panda-re friendly format(qcow2) requires non-negligible modifications to our initial setup.
This is because:

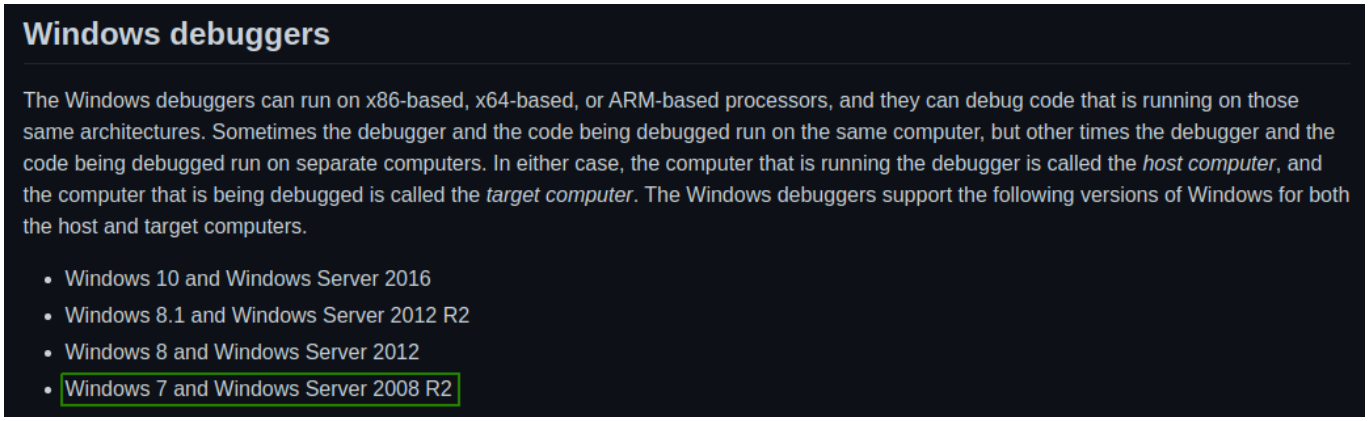
- `panda-re` introspection plugin works for a `win7x86` version
- The version used in the initial setup was a `WinServer2012R2` AMD64

From previous analysis, we're interested in a UAF bug which means that we're looking for heap allocation/freeing.

Highlights of the setup

In the optic of using `panda-re` against iMC 7.3 (E0504P04), I am going for a Windows Server 2008 because of the win7x86intro plugin.

This version of WinServer is motivated by the following:



https://github.com/MicrosoftDocs/windows-driver-docs/blob/staging/windows-driver-docs-pr/debugger/index.md

Note: the following installation/setup steps won't be as thorough as previously.

High level overview:

- Install a Win2k8(x86), not really a install because it's a VHD
- Quick test to see if `panda-system-i386` can be used to load the image/disk
  - Install iMC 7.3 (E0504P04)
- Install panda-re
  - Some configuration is required(in a first place, on the qemu side)

Win2k8(x86) installation related

Fetching a Win2k8:

https://www.microsoft.com/en-us/download/details.aspx?id=14527
Windows Server 2008 Enterprise Edition x86 (Full Install) VHD

Test if it can be used with panda-re:

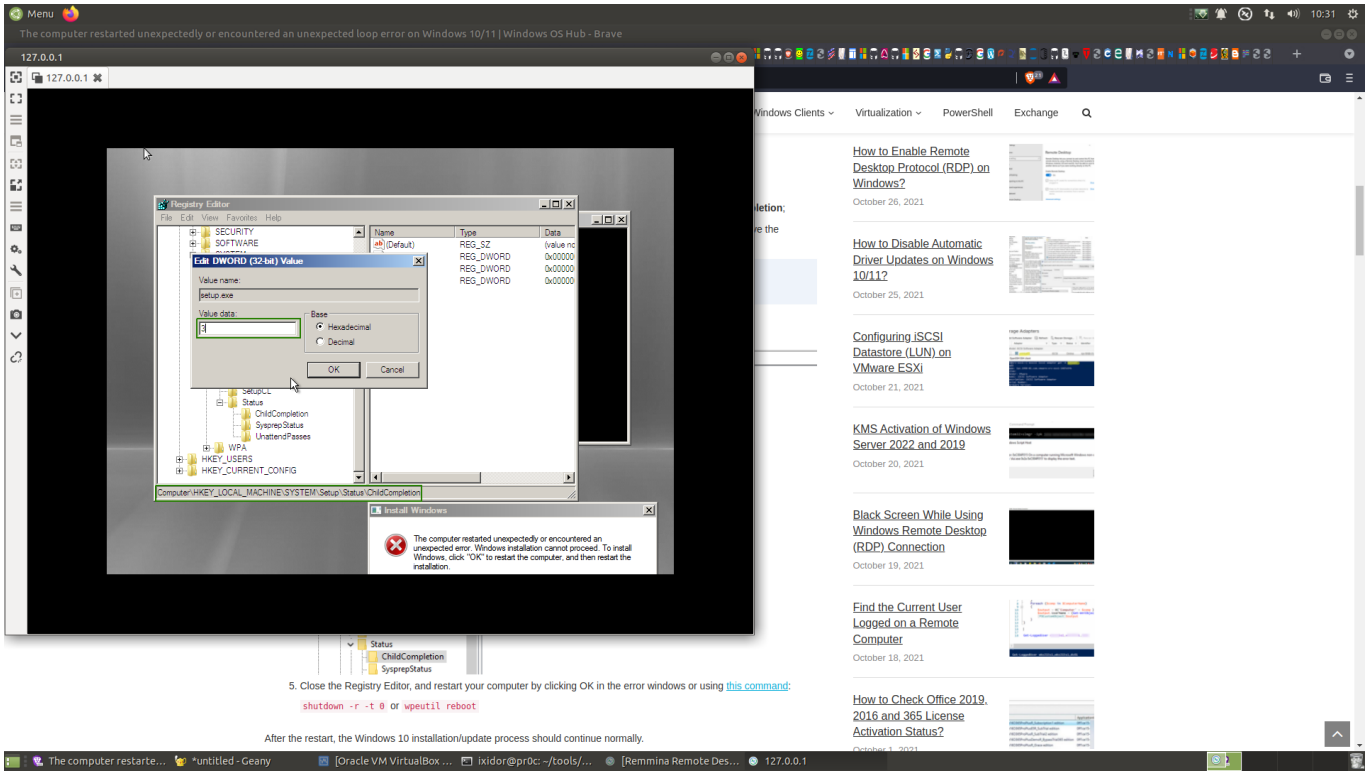
Convert the VHD to qcow2:

```
qemu-img convert -f vpc Windows2008Fullx86Ent.vhd -O qcow2 Windows2008Fullx86Ent.qcow
```

Try loading the image with `panda-system-i386`:

```
ixidor@pr0c:~/tools/panda/build/i386-softmmu$ ./panda-system-i386 -m 4096 -monitor stdio -hda ~/virt/bak/Windows2008Fullx86Ent.qcow2
```

The first load triggers the following error, which can be fixed by setting that registry key to 3:

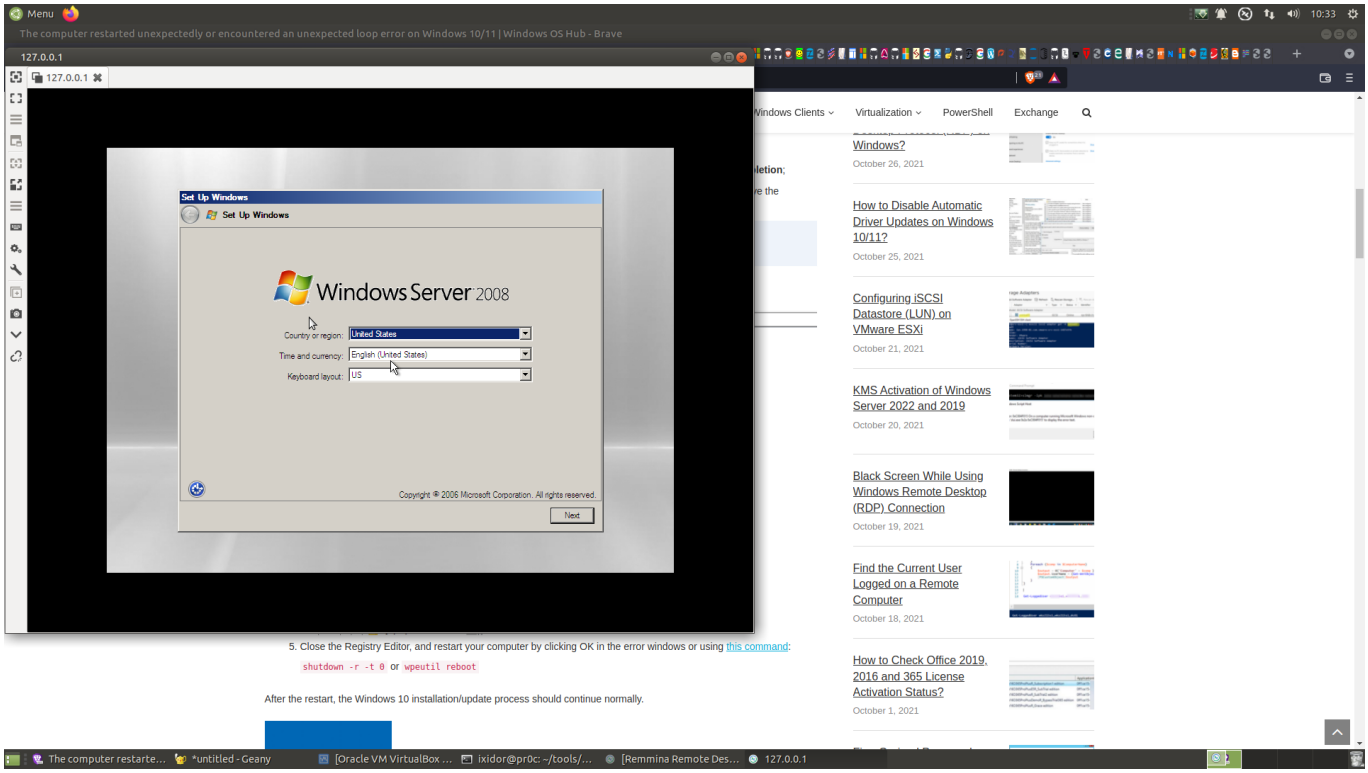


5. Close the Registry Editor, and restart your computer by clicking OK in the error windows or using this command:

```
shutdown -r -t 0
```
or
```
wpeutil reboot
```

After the restart, the Windows 10 installation/update process should continue normally.

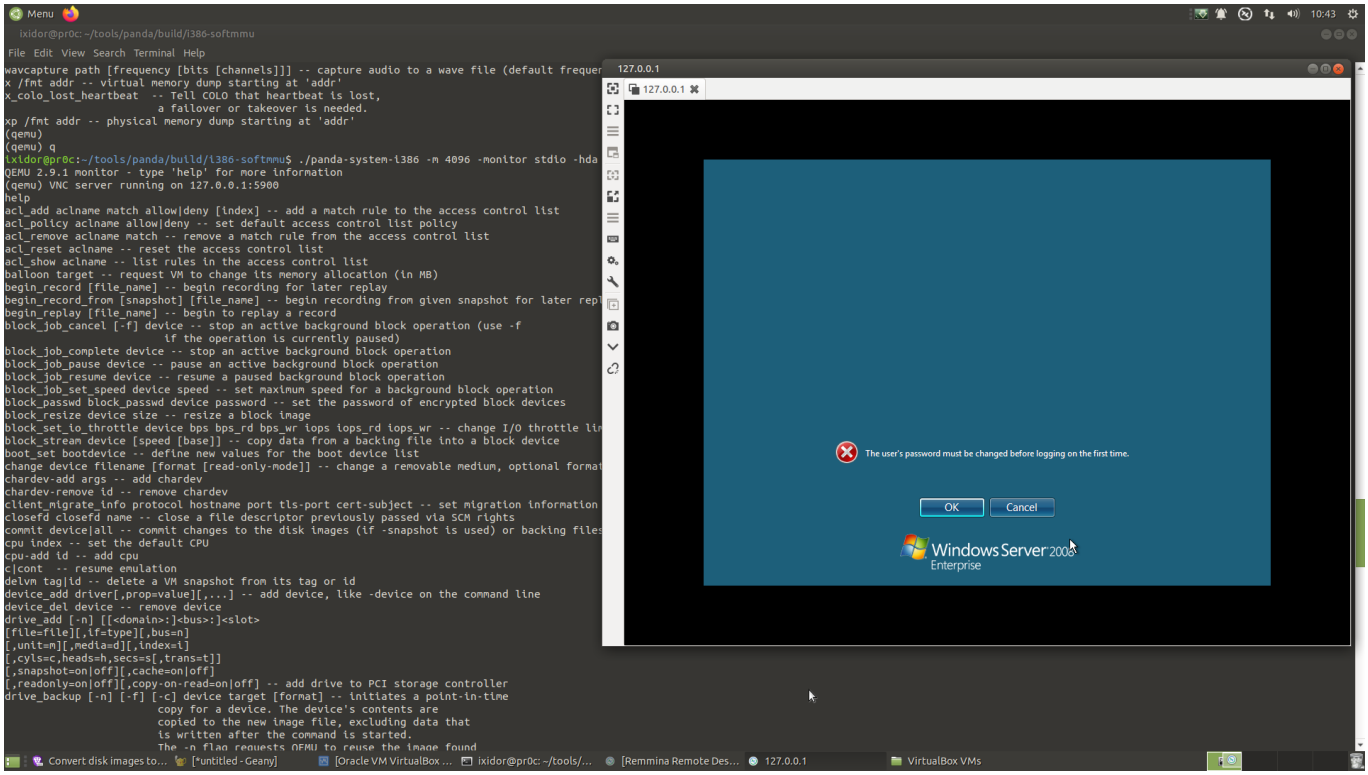
After a reboot, it should proceed smoothly:

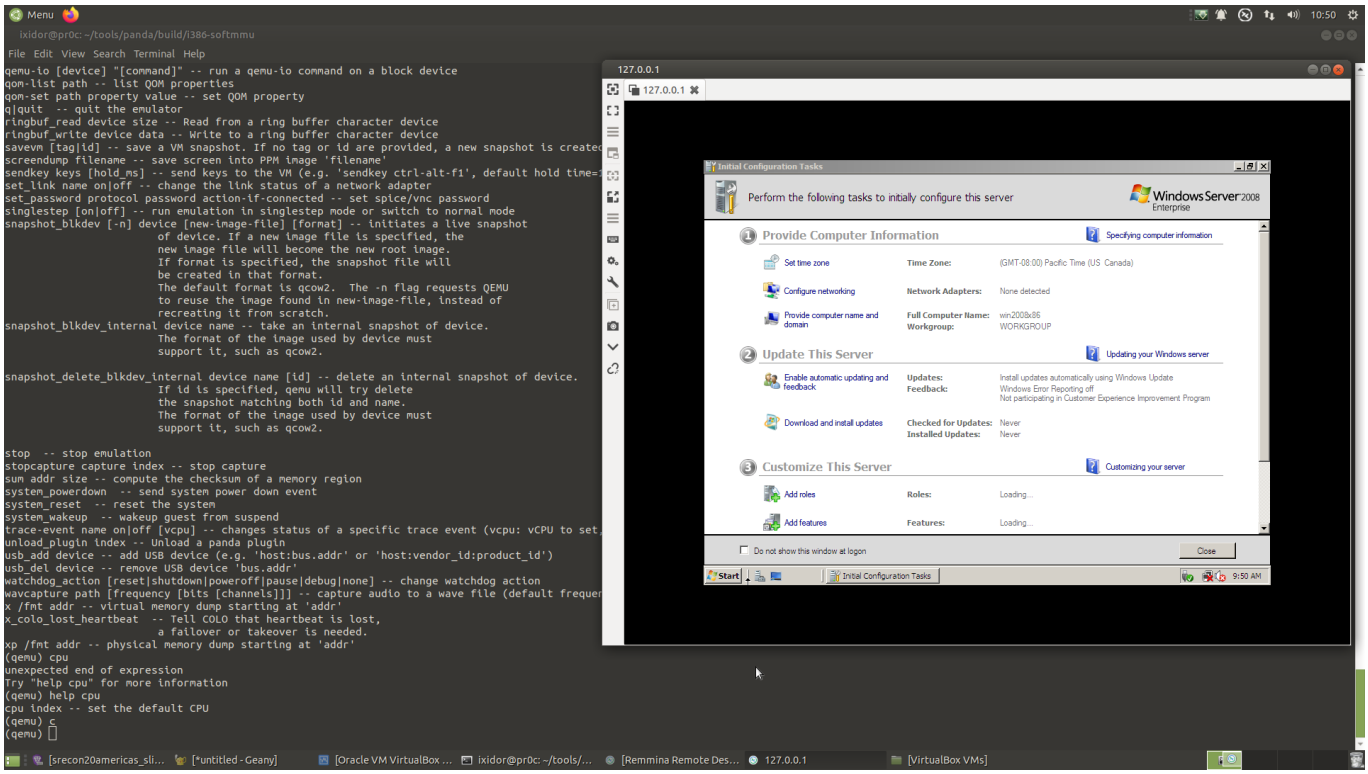


5. Close the Registry Editor, and restart your computer by clicking OK in the error windows or using this command:

```
shutdown -r -t 0
```
or
```
wpeutil reboot
```

After the restart, the Windows 10 installation/update process should continue normally.





At this point, we notice that the performance are rather slow.

To proceed with the installation of iMC, I chose to reconvert the image to a `vdi`, perform the installation in VirtualBox ; and finally, reconvert back to `qcow2` to do the analysis.

Convert qcow2 to VDI:

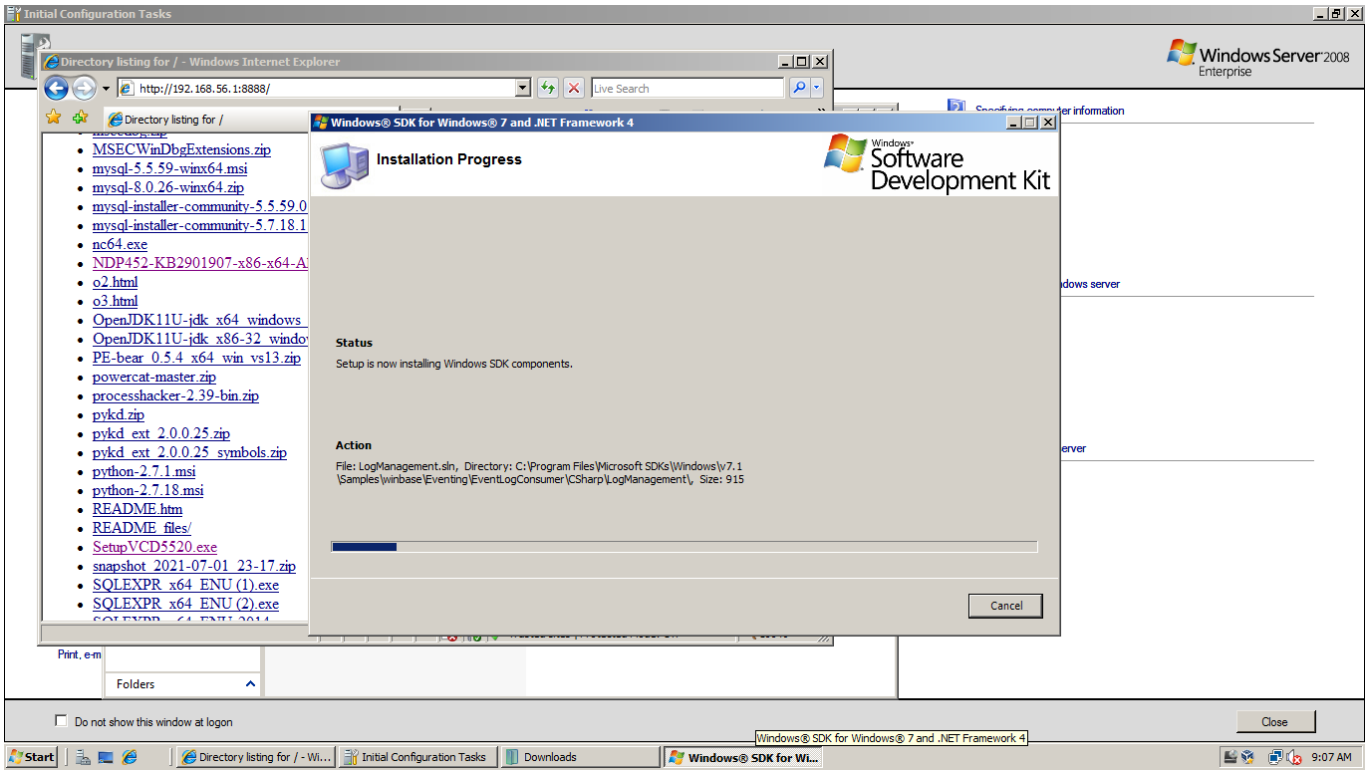
```
qemu-img convert -f qcow2 Windows2008Fullx86Ent.qcow2 -O vdi
Windows2008Fullx86Ent.vdi
```

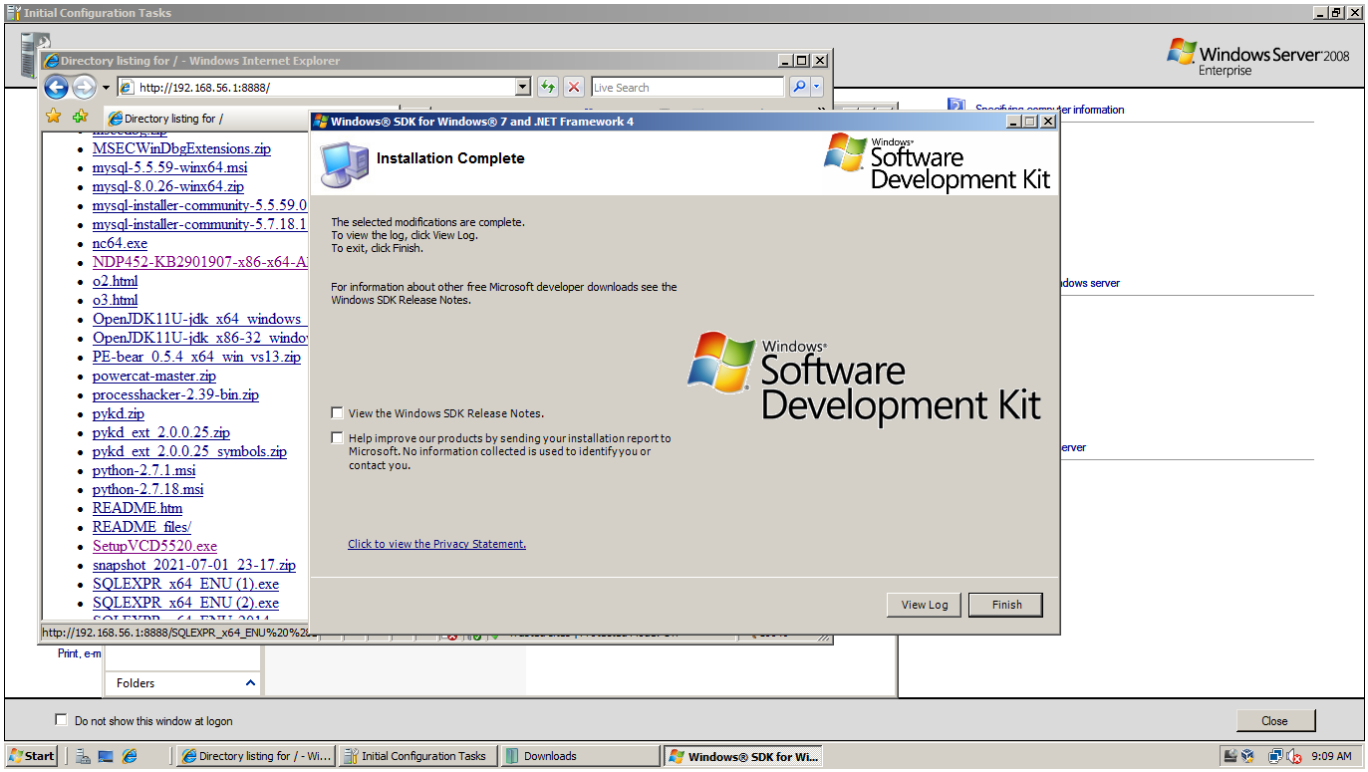
We still want windbg to see if we observe the same behavior as previously.

To get windbg:

https://www.elby.ch/en/products/vcd.html
VirtualCloneDrive

https://www.microsoft.com/en-us/download/details.aspx?id=8442
Microsoft Windows SDK for Windows 7 and .NET Framework 4 (ISO)



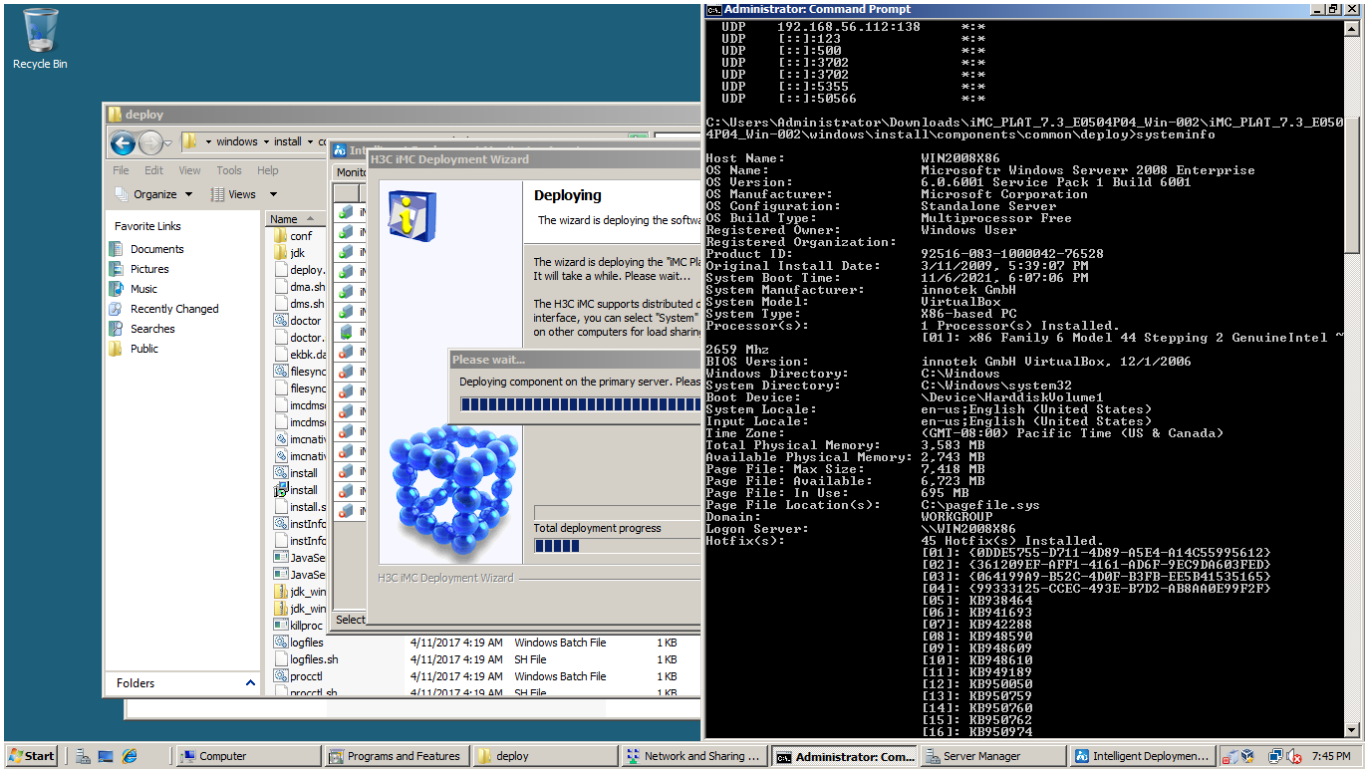


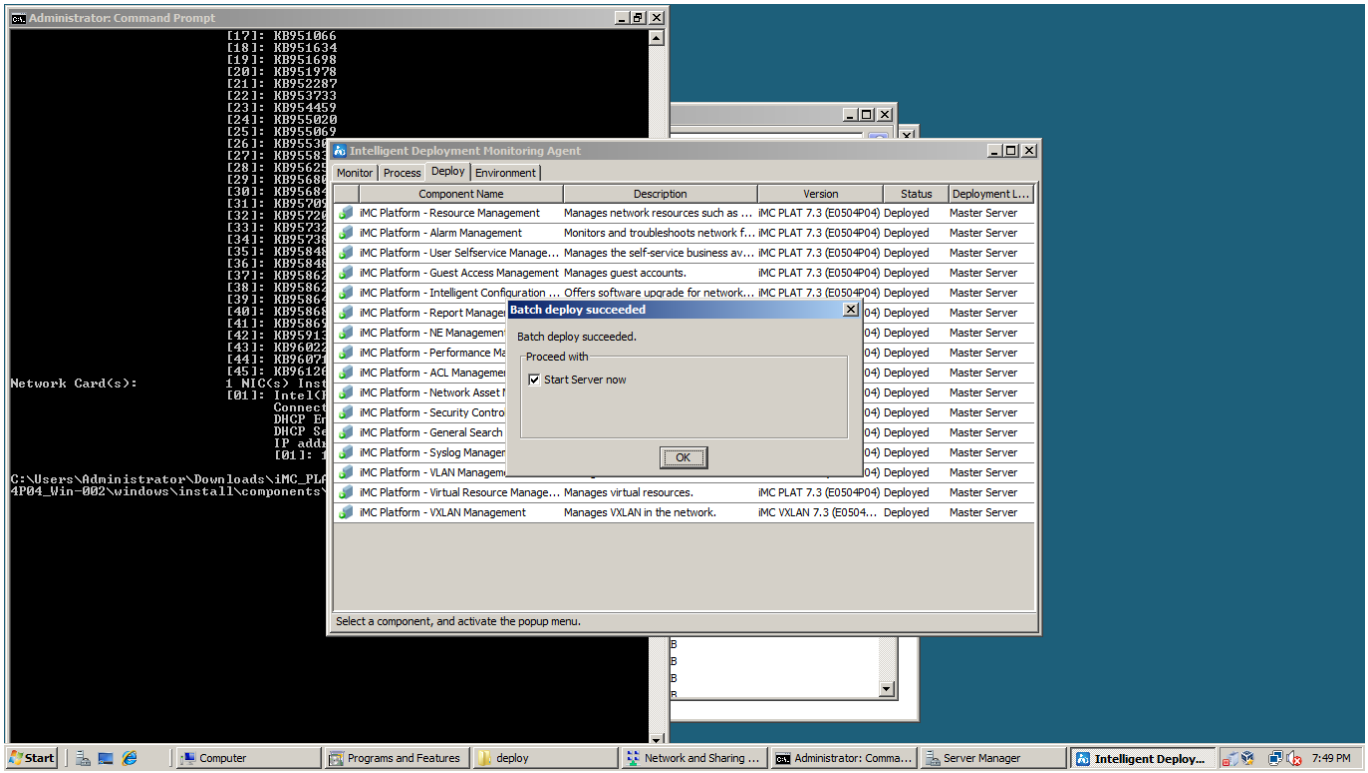
https://www.microsoft.com/en-us/download/details.aspx?id=11800
Windows Driver Kit Version 7.1.0

iMC 7.3 (E0504P04)

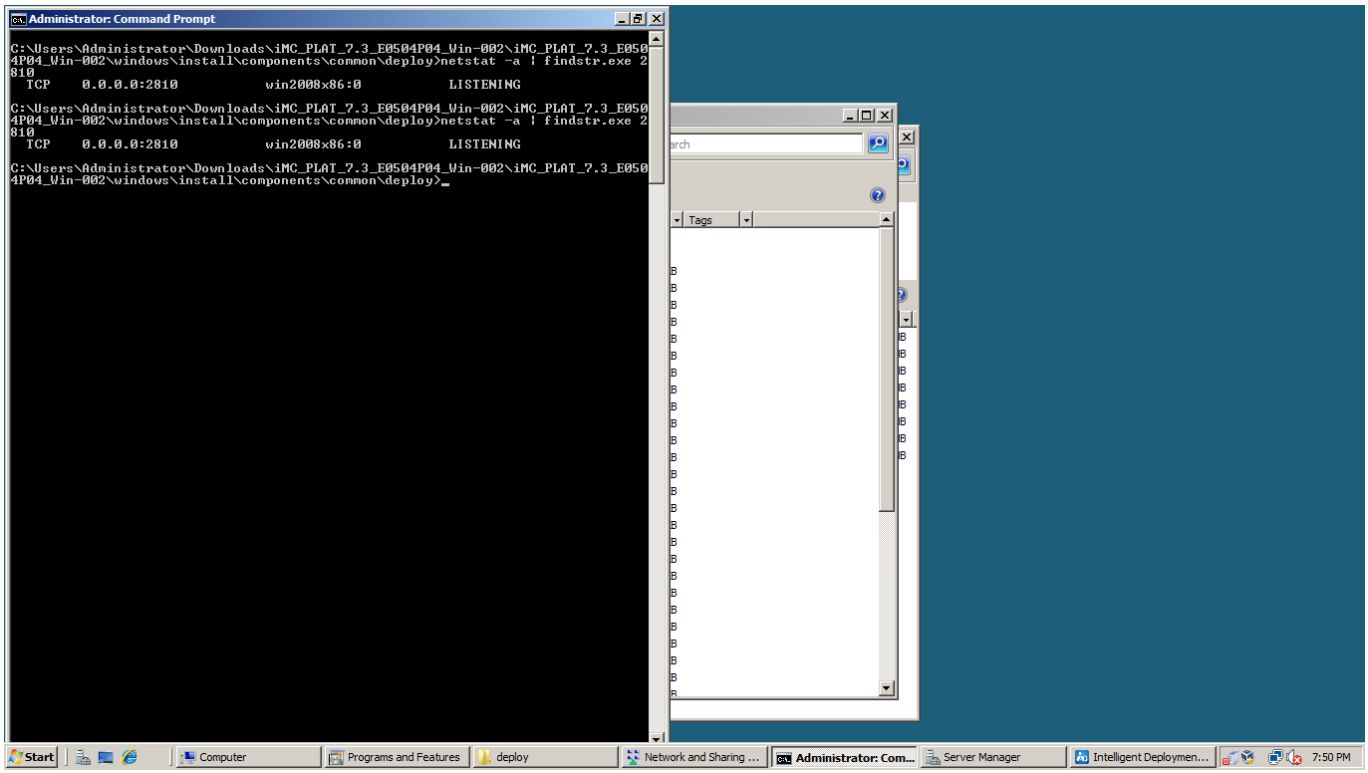
The installation of iMC is almost identical to the Win2k12R2 version except that we're using the SQLServerExpress2008R2 database software provided in one of the iMC archive.

iMC deployment final phases:





dbman.exe is running:



Finally, we convert vdi to qcow2:

```
qemu-img convert -f vdi Windows2008Fullx86Ent.vdi -O qcow2 Windows2008Fullx86Ent.qcow2
```

Note: it is likely that we could avoid converting from one format to another but that path hasn't been explored.

Armed with `Windows2008Fullx86Ent.qcow2`, we can feed it to `panda-system-i386`.

panda-re

panda-re -- install

Installing panda-re on Ubuntu 20.04.3 LTS:

```
$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 20.04.3 LTS
Release:    20.04
Codename:   focal
```

Clone:
`git clone https://github.com/panda-re/panda`

I made two pull requests to the main branch because of a small execution flow and permission issue in `install_ubuntu.sh`.

There could be a possible third pull request but I assume that people perform an `apt-get update && apt-get upgrade -y` before launching the `install_ubuntu.sh` script.

The installation takes a while(getting dependencies + compilation) and requires space up to twenty gigas ; this is a rough estimate.

panda-re -- config

In order to be able to load/boot/emulate the `Windows2008Fullx86Ent.qcow2` machine, a number of arguments for `panda-system-i386` are needed.

Version:

```
ixidor@pr0c:~/tools/panda/build/i386-softmmu$ ./panda-system-i386 -version
QEMU emulator version 2.9.1 (-dirty)
Build date Nov  5 2021
Copyright (c) 2003-2017 Fabrice Bellard and the QEMU Project developers
```

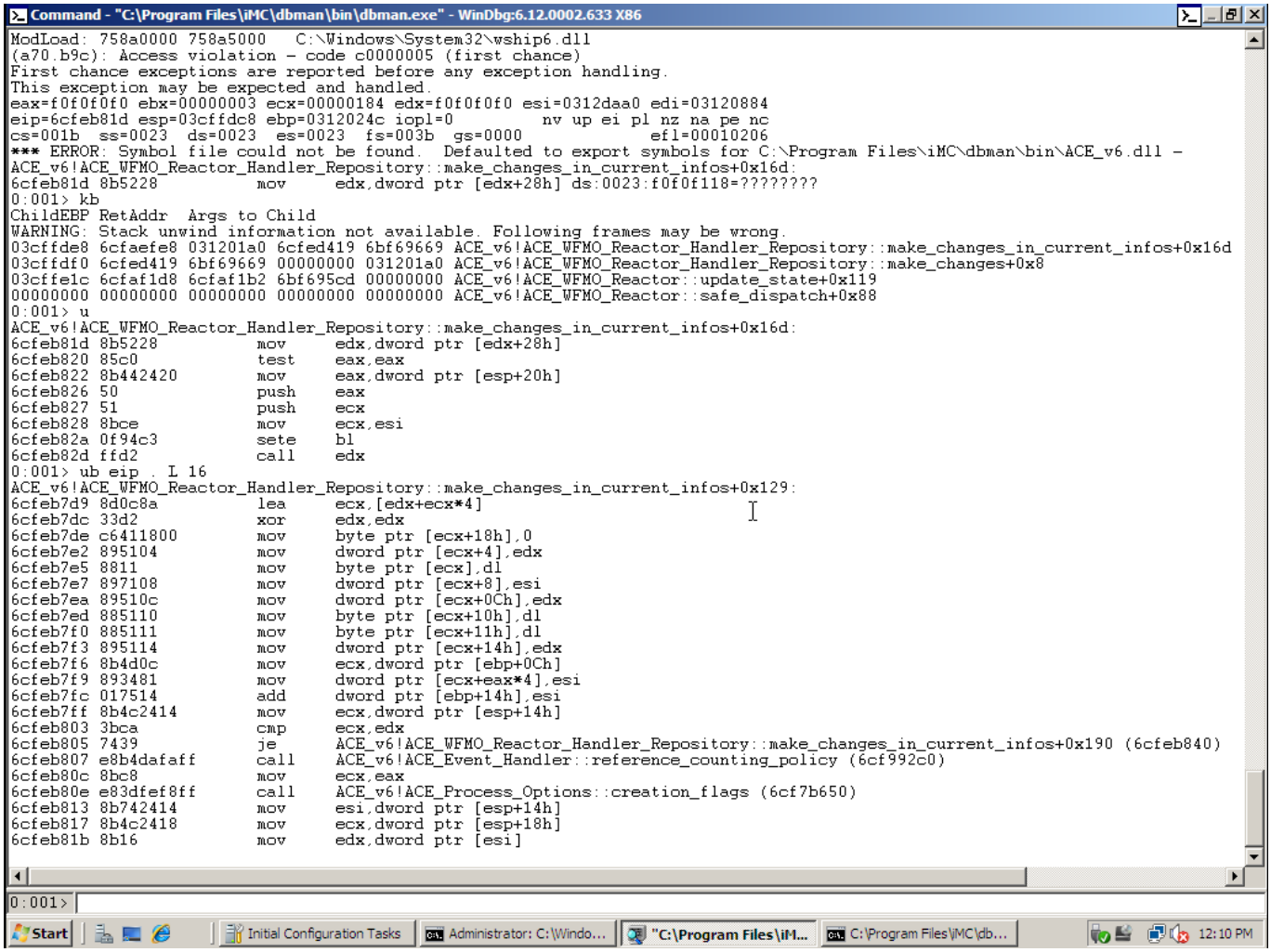
I am purposely not going into the details of this command line:

`ixidor@pr0c:~/tools/panda/build/i386-softmmu$ sudo ./panda-system-i386 -enable-kvm -m 4096 -net nic -net tap,ifname=tap0,script=no -usbdevice tablet -monitor stdio -vga virtio -hda ~/virt/bak/Windows2008Fullx86Ent.qcow2`

- `-enable-kvm`: use kvm acceleration, increases performance
- `-m`: dedicated memory
- `-net nic -net tap,ifname=tap0,script=no`: adding a network card to the guest and communicating through a tap interface(L2), avoid calling `panda-bridge-helper`; there is at least one cleaner way to do this but this works
- `-usbdevice tablet`: sync mouse cursors between the host and the guest
- `-vga virtio`: enable the possibility to increase screen resolution

Looking for the bug through panda/qemu hypervisor

Trigger av in `dbman.exe` in windbg within the `qemu` environment:



The result is in the same vein of experiments conducted in the win2k12R2 version.
At this point, except the hypervisor and the MS Windows version nothing has changed.

We should be able to make use of some `panda` magic.

Other links

http://woshub.com/windows-install-error-computer-restarted-unexpectedly/

https://techpiezo.com/linux/convert-disk-images-to-various-formats-using-qemu-img/

https://web.archive.org/web/20210125093418/http://download.microsoft.com/download/2/6/1/261fca42-22c0-4f91-9451-0e0f2e08356d/Windows6.0-KB942288-v2-x86.msu
Windows Installer 4.5

QEMU's new -nic command line option - QEMU

Looking at Java Classes

In this context, the iMC software makes use of both Open Source:

```
C:\Program Files\iMC\server\bin\
```

```
ACE.dll
ACE_SSL.dll
ACE_v6.dll
```

even though compiled, it probably refers to the following software stack:

The ADAPTIVE Communication Environment

http://www.dre.vanderbilt.edu/~schmidt/ACE-overview.html

and probably closed source software:

```
C:\Program Files\iMC\anti-vm\bin\anti-vm.exe
```

```
md5sum
dfffcea7c76052be489028287acbee31  anti-vm.exe
```

Java Classes

In this greybox setting, we can look at *jars*, e.g.:

```
C:\Program Files\iMC\deploy\deploy.jar
```

Using tools such as `gadgetinspector`:

Tool description taken from:

```
https://github.com/JackOfMostTrades/gadgetinspector
```

Gadget Inspector

This project inspects Java libraries and classpaths for gadget chains. Gadgets chains are used to construct exploits for deserialization vulnerabilities. By automatically discovering possible gadgets chains in an application's classpath penetration testers can quickly construct exploits and application security engineers can assess the impact of a deserialization vulnerability and prioritize its remediation.

This project was presented at Black Hat USA 2018. Learn more about it there! (Links pending)

DISCLAIMER: This project is alpha at best. It needs tests and documentation added. Feel free to help by adding either!

Running the tool:

```
┌──(kali㉿kali)-[~/gadgetinspector]
└─$ nl gadget-chains.txt
     1  org/apache/commons/lang/enums/Enum.equals(Ljava/lang/Object;)Z (1)
     2    org/apache/commons/lang/enums/Enum.getNameInOtherClassLoader(Ljava/lang/Object;)Ljava/lang/String; (1)
     3    java/lang/reflect/Method.invoke(Ljava/lang/Object;[Ljava/lang/Object;)Ljava/lang/Object; (0)

     4  javax/security/auth/kerberos/KerberosTicket.equals(Ljava/lang/Object;)Z (1)
     5    javax/security/auth/kerberos/KerberosTicket.getRenewTill()Ljava/util/Date; (0)
     6    java/util/Date.clone()Ljava/lang/Object; (0)
     7    org/jfree/chart/plot/CategoryPlot.clone()Ljava/lang/Object; (0)
     8    org/jfree/util/ObjectUtilities.clone(Ljava/lang/Object;)Ljava/lang/Object; (0)
     9    java/lang/reflect/Method.invoke(Ljava/lang/Object;[Ljava/lang/Object;)Ljava/lang/Object; (0)

    10  org/springframework/aop/framework/JdkDynamicAopProxy.invoke(Ljava/lang/Object;Ljava/lang/reflect/Method;[Ljava/lang/Object;)Ljava/lang/Object; (0)
    11    org/springframework/aop/target/ThreadLocalTargetSource.getTarget()Ljava/lang/Object; (0)
    12    edu/emory/mathcs/backport/java/util/concurrent/ConcurrentSkipListSet$ConcurrentSkipListSubSet.add(Ljava/lang/Object;)Z (0)
    13    edu/emory/mathcs/backport/java/util/concurrent/ConcurrentSkipListMap$ConcurrentSkipListSubMap.put(Ljava/lang/Object;Ljava/lang/Object;)Ljava/lang/Object; (0)
    14    edu/emory/mathcs/backport/java/util/concurrent/ConcurrentSkipListMap$ConcurrentSkipListSubMap.checkKey(Ljava/lang/Object;)V (0)
    15    edu/emory/mathcs/backport/java/util/concurrent/ConcurrentSkipListMap$ConcurrentSkipListSubMap.inHalfOpenRange(Ljava/lang/Object;)Z (0)
    16    edu/emory/mathcs/backport/java/util/concurrent/ConcurrentSkipListMap.inHalfOpenRange(Ljava/lang/Object;Ljava/lang/Object;Ljava/lang/Object;)Z (2)
    17    edu/emory/mathcs/backport/java/util/concurrent/ConcurrentSkipListMap.compare(Ljava/lang/Object;Ljava/lang/Object;)I (2)
    18    org/apache/commons/lang/enums/ValuedEnum.compareTo(Ljava/lang/Object;)I (1)
    19    org/apache/commons/lang/enums/ValuedEnum.getValueInOtherClassLoader(Ljava/lang/Object;)I (1)
    20    java/lang/reflect/Method.invoke(Ljava/lang/Object;[Ljava/lang/Object;)Ljava/lang/Object; (0)

    21  org/apache/commons/lang/enum/Enum.equals(Ljava/lang/Object;)Z (1)
    22    org/apache/commons/lang/enum/Enum.getNameInOtherClassLoader(Ljava/lang/Object;)Ljava/lang/String; (1)
    23    java/lang/reflect/Method.invoke(Ljava/lang/Object;[Ljava/lang/Object;)Ljava/lang/Object; (0)

    24  javax/security/auth/kerberos/KerberosTicket.equals(Ljava/lang/Object;)Z (1)
    25    javax/security/auth/kerberos/KerberosTicket.getRenewTill()Ljava/util/Date; (0)
    26    java/util/Date.clone()Ljava/lang/Object; (0)
    27    org/jfree/data/KeyToGroupMap.clone()Ljava/lang/Object; (0)
    28    org/jfree/data/KeyToGroupMap.clone(Ljava/lang/Object;)Ljava/lang/Object; (0)
    29    java/lang/reflect/Method.invoke(Ljava/lang/Object;[Ljava/lang/Object;)Ljava/lang/Object; (0)
```

As stated in the project's github page, this doesn't mean that their actually is a vulnerability. `gadgetinspector` looks for gadgets inside Java classes and builds up a chain. The question if this chain can be `triggered` needs to be tested in an operational environment.

Links and References

https://www.zerodayinitiative.com/advisories/ZDI-17-836/

https://www.cvedetails.com/cve/CVE-2017-12561/

https://github.com/pwnslinger/exploit-repo/tree/master/CVE-2017-12561

https://raw.githubusercontent.com/googleprojectzero/0days-in-the-wild/main/0day-RCAs/template.md

https://medium.com/tenable-techblog/hpe-imc-hunting-the-hunted-2619889802ab

https://deepsec.net/docs/Slides/2012/DeepSec_2012_Rosario_Valotta_-_Taking_Browsers_Fuzzing_to_the_next_(DOM)_Level.pdf

https://www.itu.int/rec/T-REC-X.690

https://www.cybersecurity-help.cz/vdb/SB2017100402

Papers

Finding the needle in the heap: combining binary analysis techniques to trigger use-after-free
https://tel.archives-ouvertes.fr/tel-01681707v2/document

https://hal.univ-grenoble-alpes.fr/hal-01721539/document

Document Editing

https://obsidian.md/

https://typora.io/