

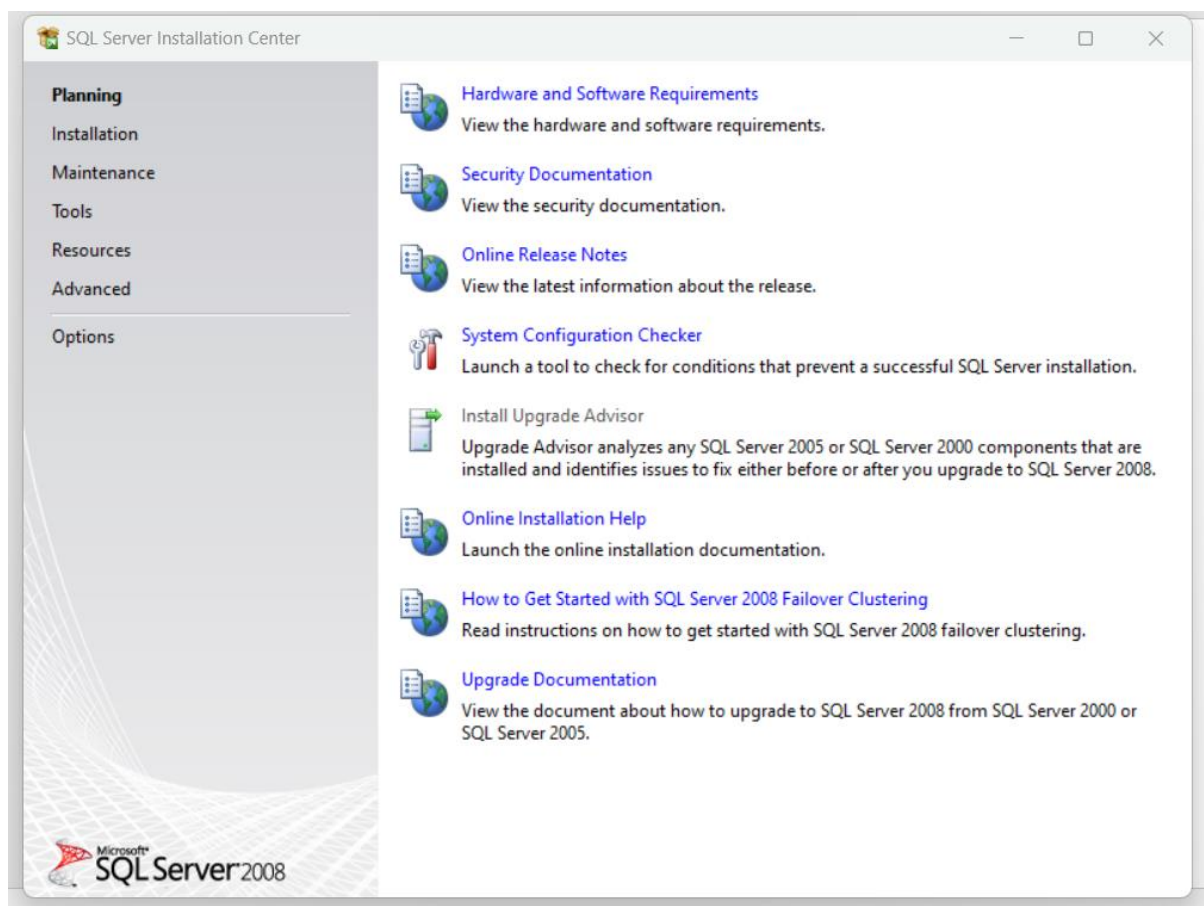
Assignment 2

CB.SC.P2CYS23017

DVTA Setup - DVTA - Part 1 - Setup (parsiya.net)

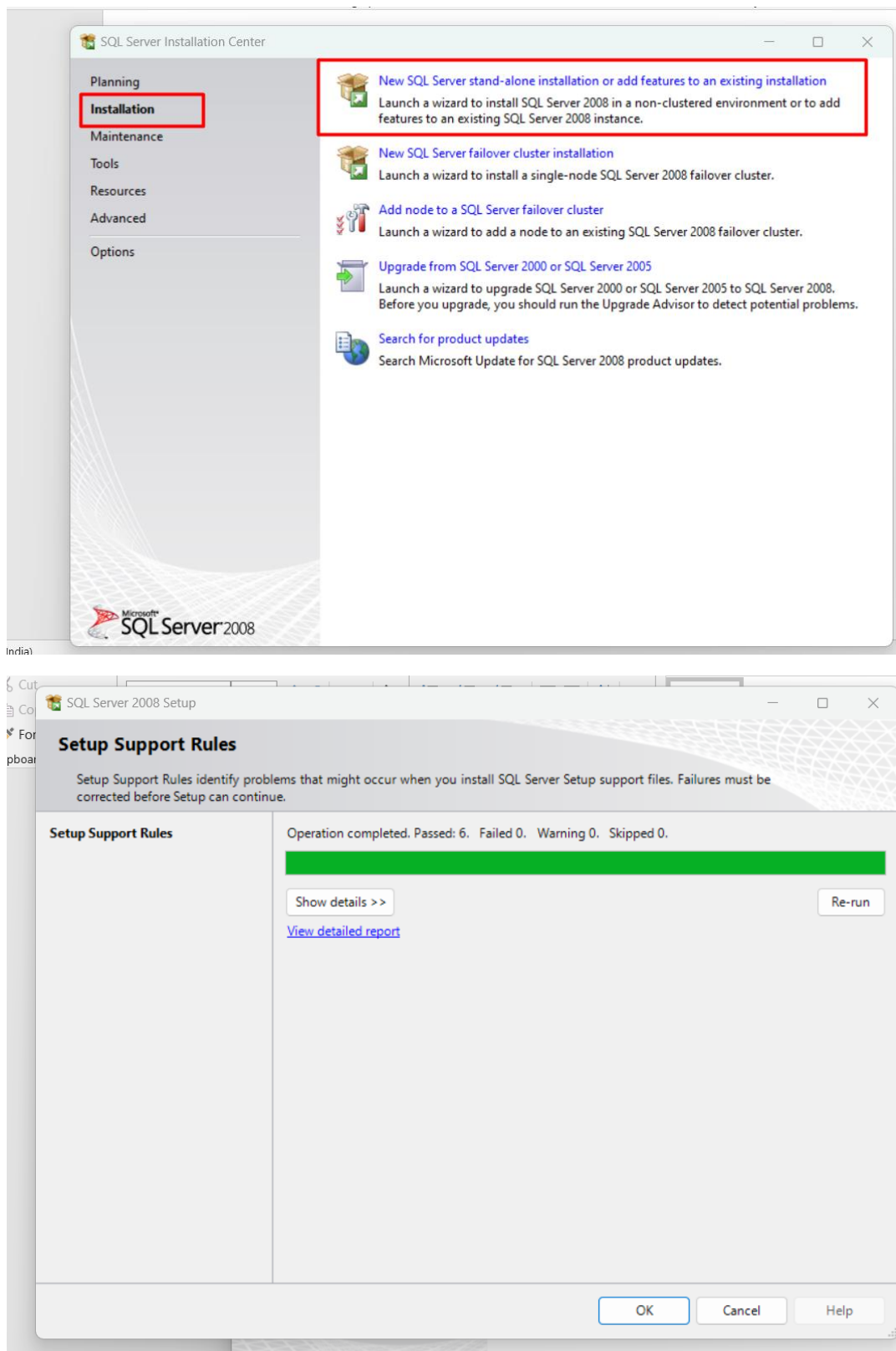
Use CFF Framework to analyse custom DVTA.exe created above.

To Start with the DVTA, first we need to setup SQL Server 2008



Assignment 2

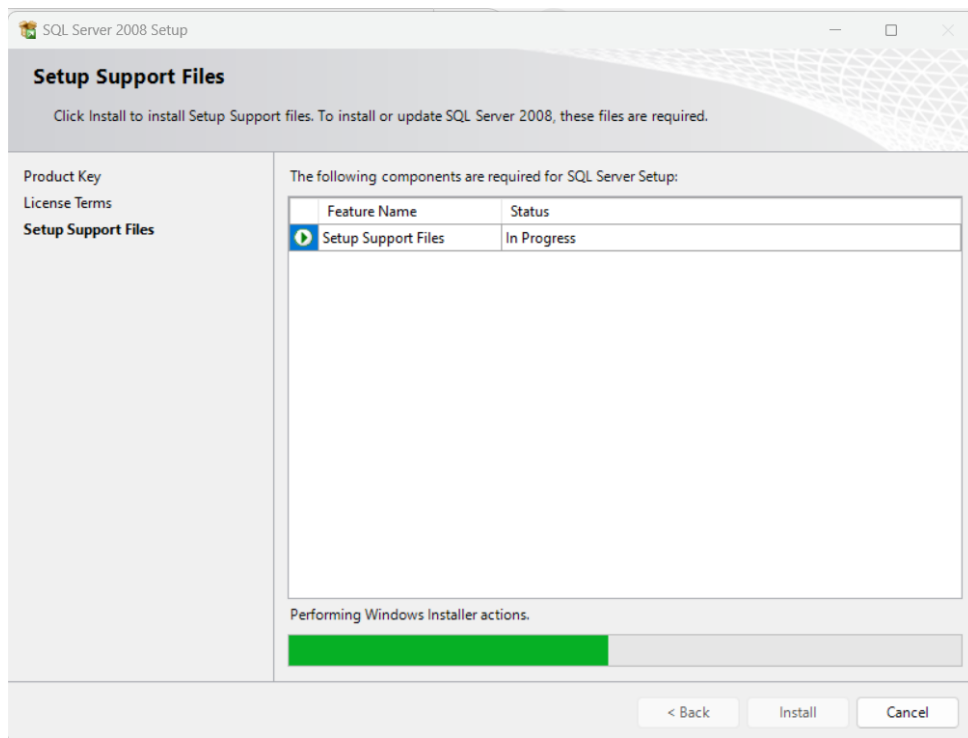
CB.SC.P2CYS23017



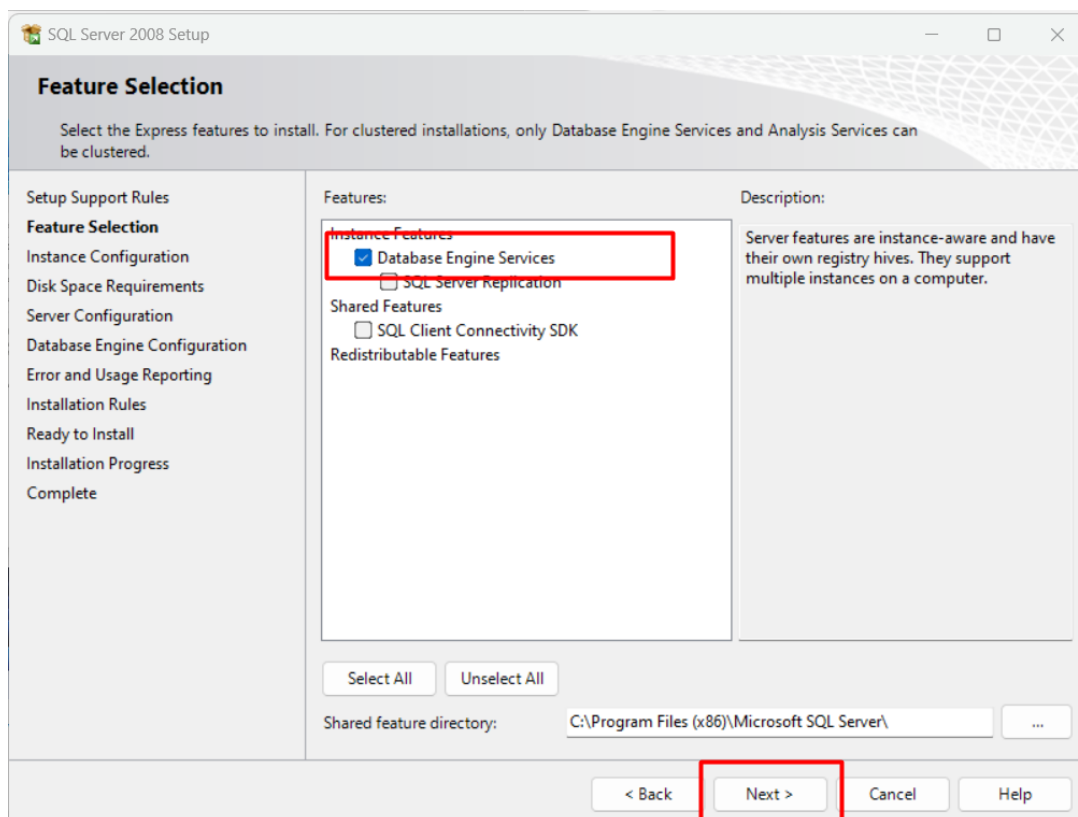
We have to setup support files

Assignment 2

CB.SC.P2CYS23017



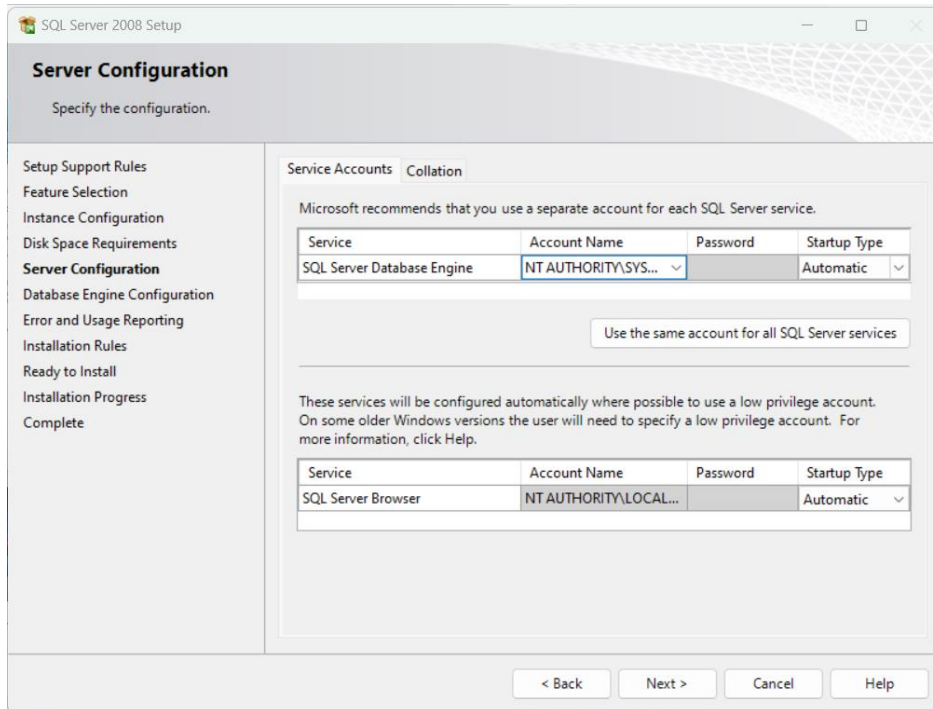
To Add Database Engine



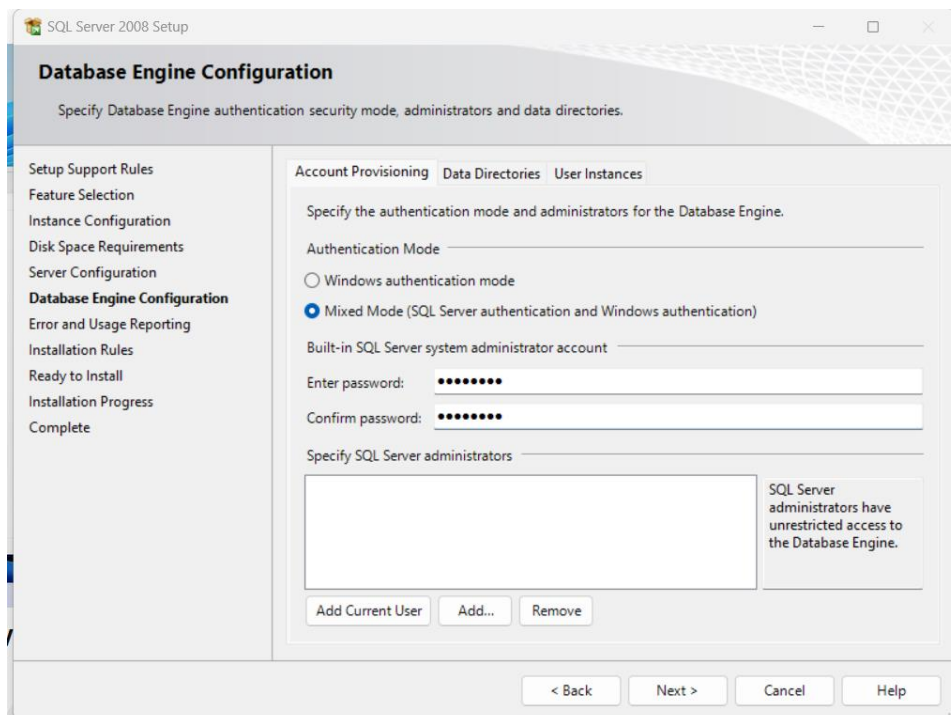
Assignment 2

CB.SC.P2CYS23017

Now we have to configure server



Add a password as a configuration of Server

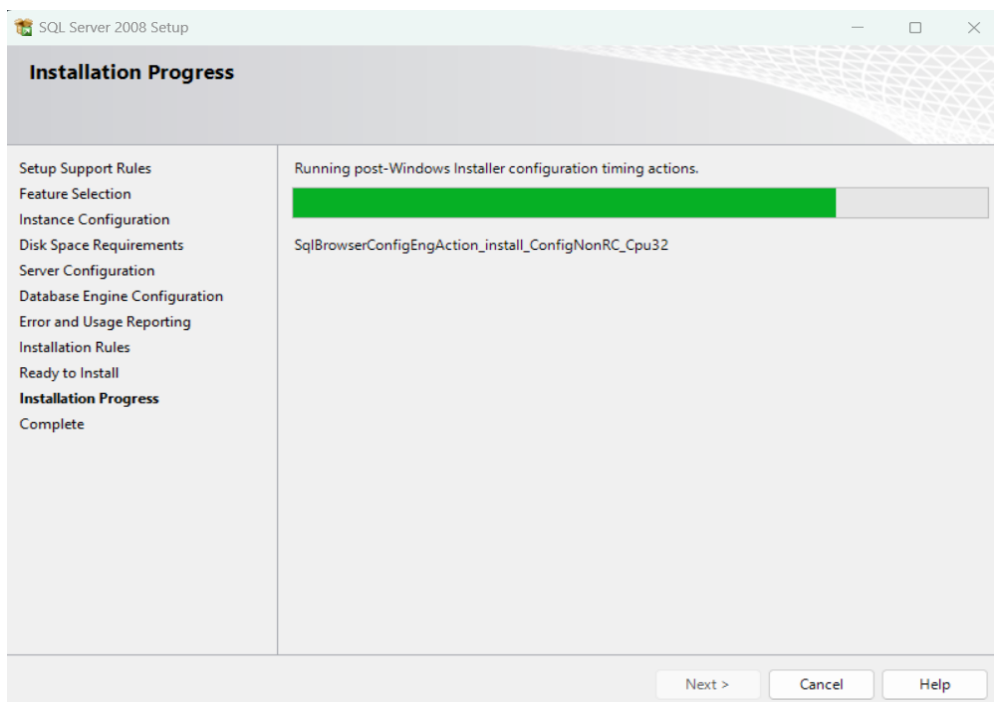
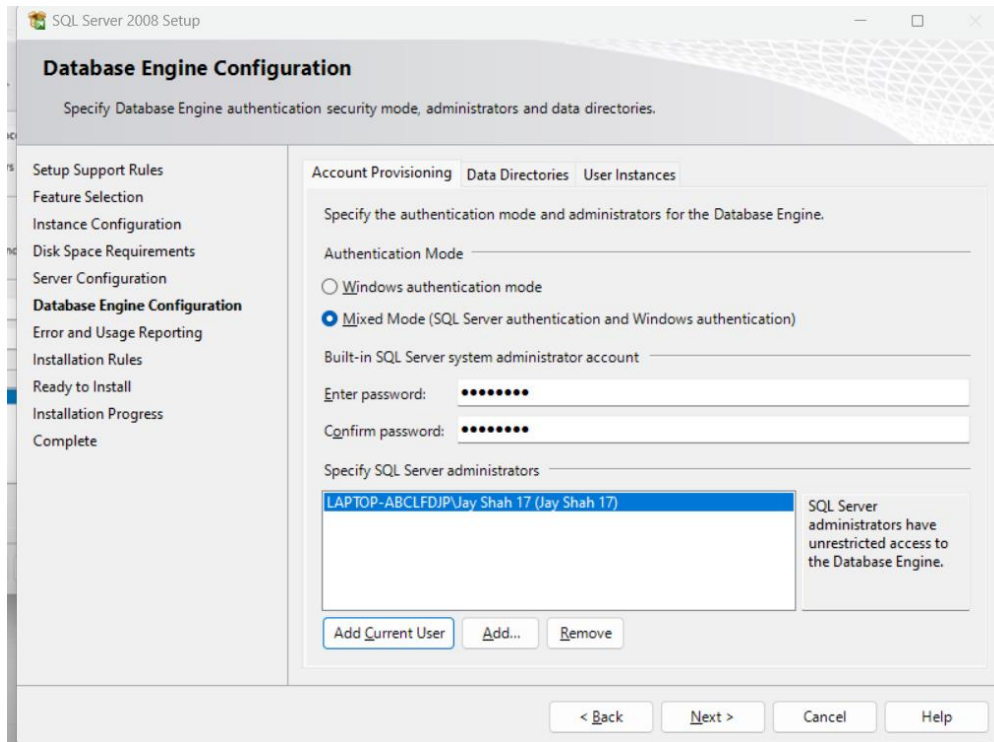


Assignment 2

CB.SC.P2CYS23017

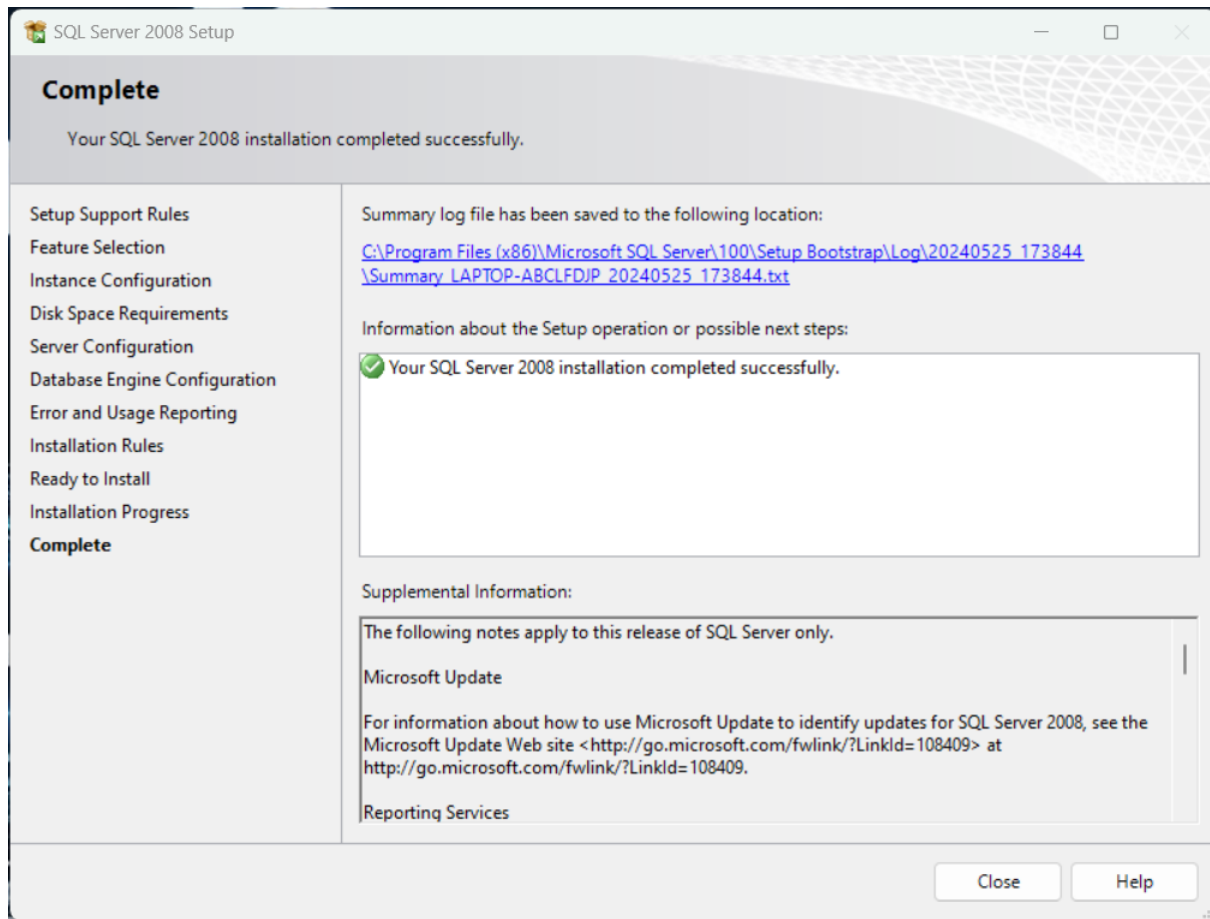
Password Should be – P@ssw0rd

- We need to add users SQL Server Administrator



Assignment 2

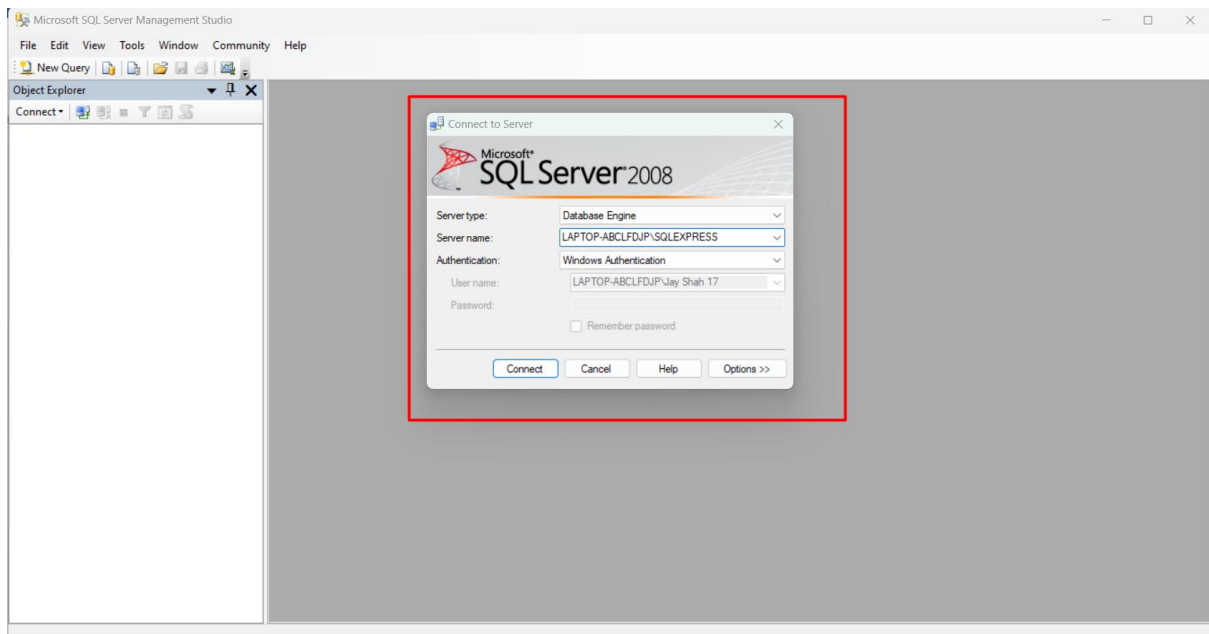
CB.SC.P2CYS23017



- We need to start SQL Server 2008

Assignment 2

CB.SC.P2CYS23017



Create New Database

Assignment 2

CB.SC.P2CYS23017

New Database

Select a page: General, Options, Filegroups

Script Help

Database name: DVTA

Owner: <default>

☒ Use full-text indexing

Database files:

Logical Name	File Type	Filegroup	Initial Size (MB)	Autogrowth
DVTA	Rows ...	PRIMARY	2	By 1 MB, unrestricted growth
DVTA_log	Log	Not Applicable	1	By 10 percent, unrestricted growth

Connection:

Server: LAPTOP-ABCLFDJP\SQLEXPRESS

Connection: LAPTOP-ABCLFDJP\Jay Shah

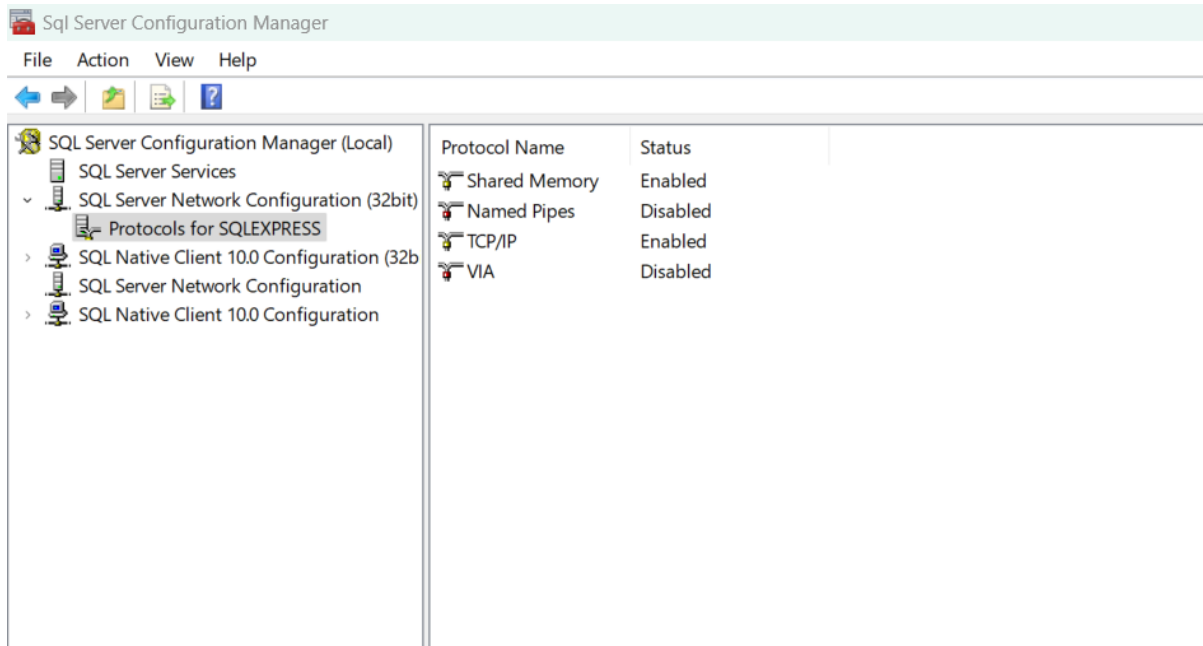
[View connection properties](#)

Progress: Ready

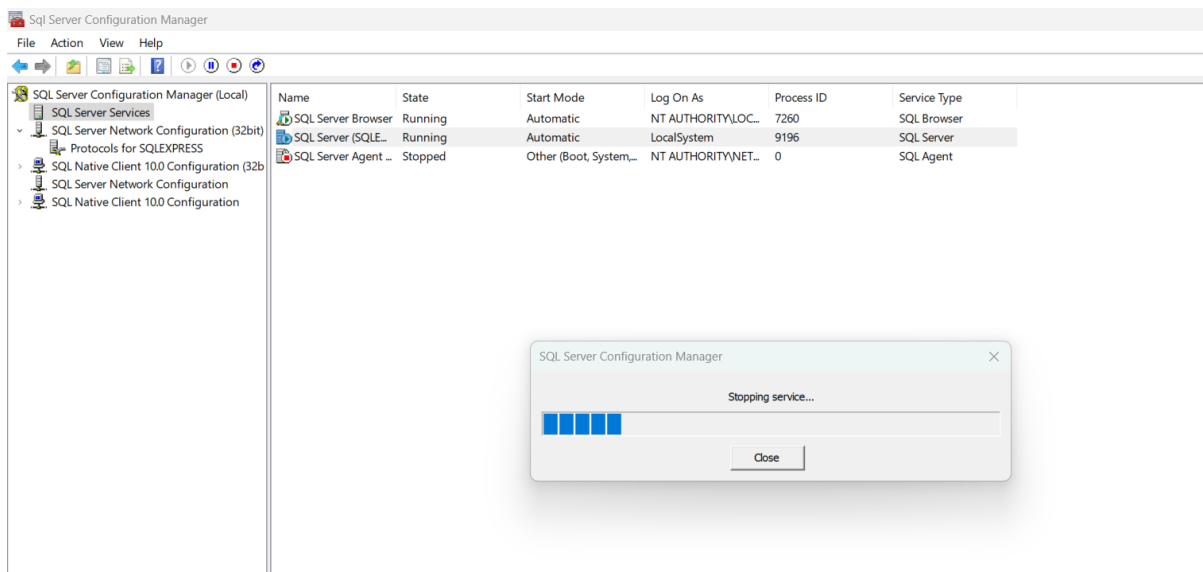
Add Remove OK Cancel

Assignment 2

CB.SC.P2CYS23017



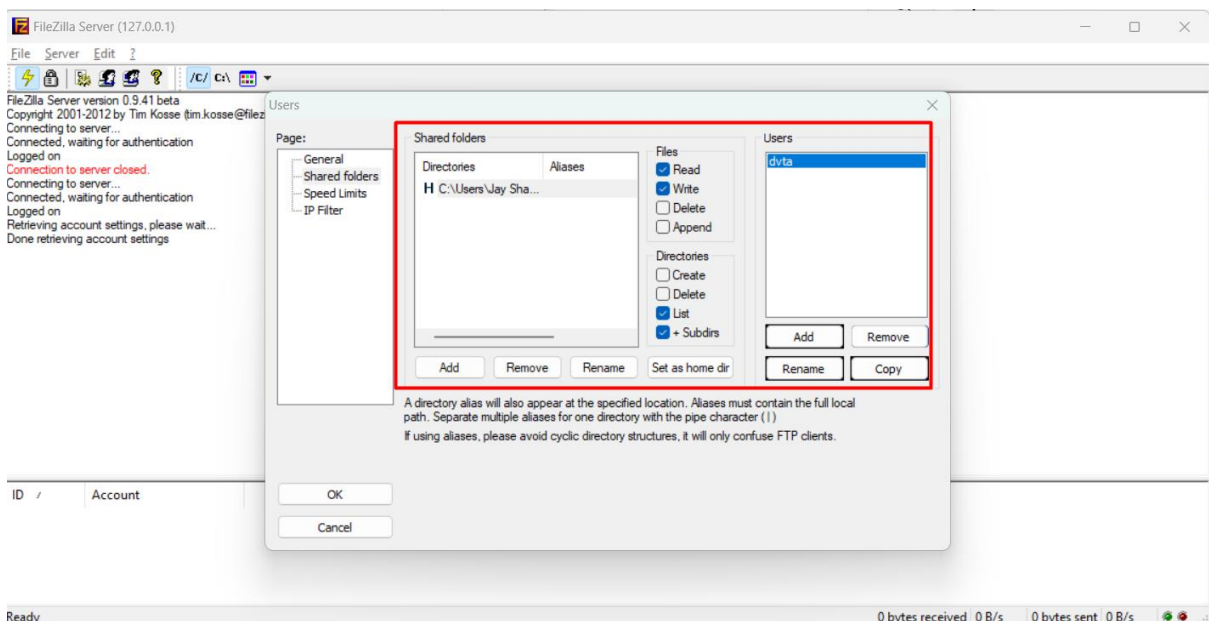
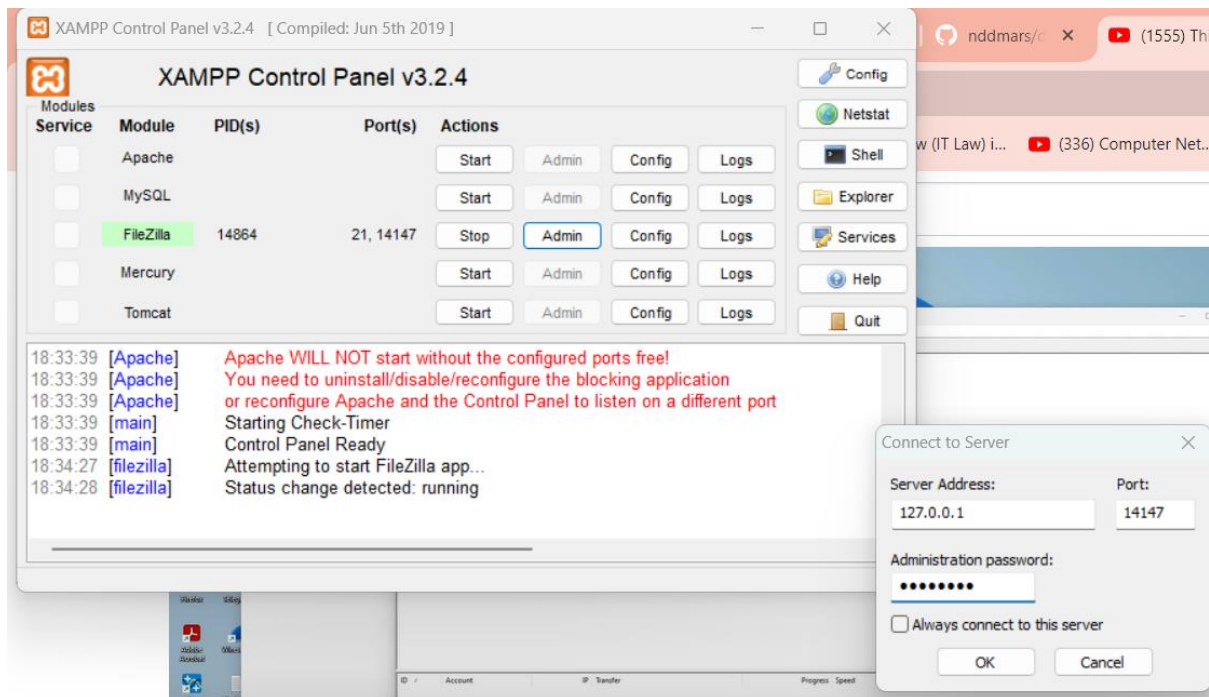
- Restart SQL Server



- Start a Filezilla Server in your computer
- Going to Admin Panel

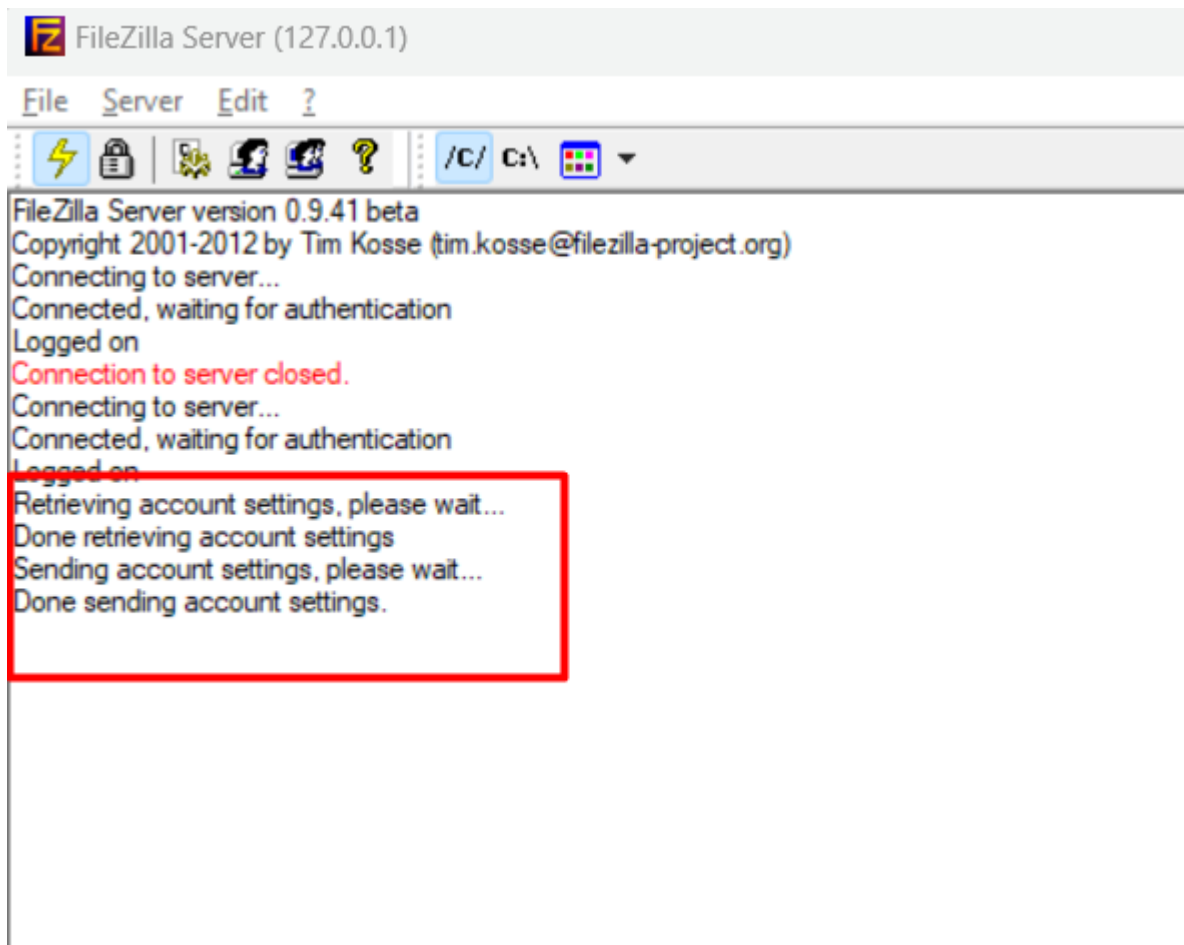
Assignment 2

CB.SC.P2CYS23017



Assignment 2

CB.SC.P2CYS23017

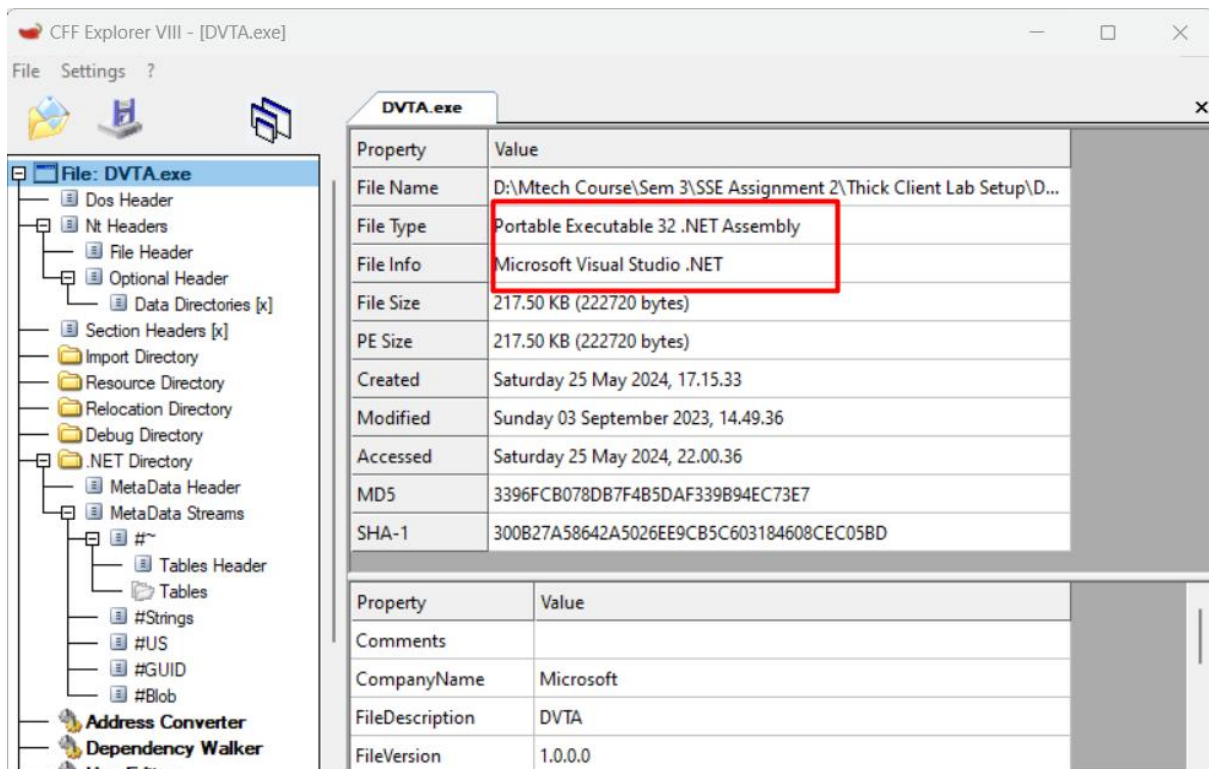


1) Identify the Application architecture, languages and frameworks used?

- Upon opening the dvta.exe in CFF-explorer, we can identify the following information
- Architecture – 32 bit & 2 tier [Since it communicates with the database.]
- Languages used - .NET Assembly
- Frameworks - .NET framework

Assignment 2

CB.SC.P2CYS23017

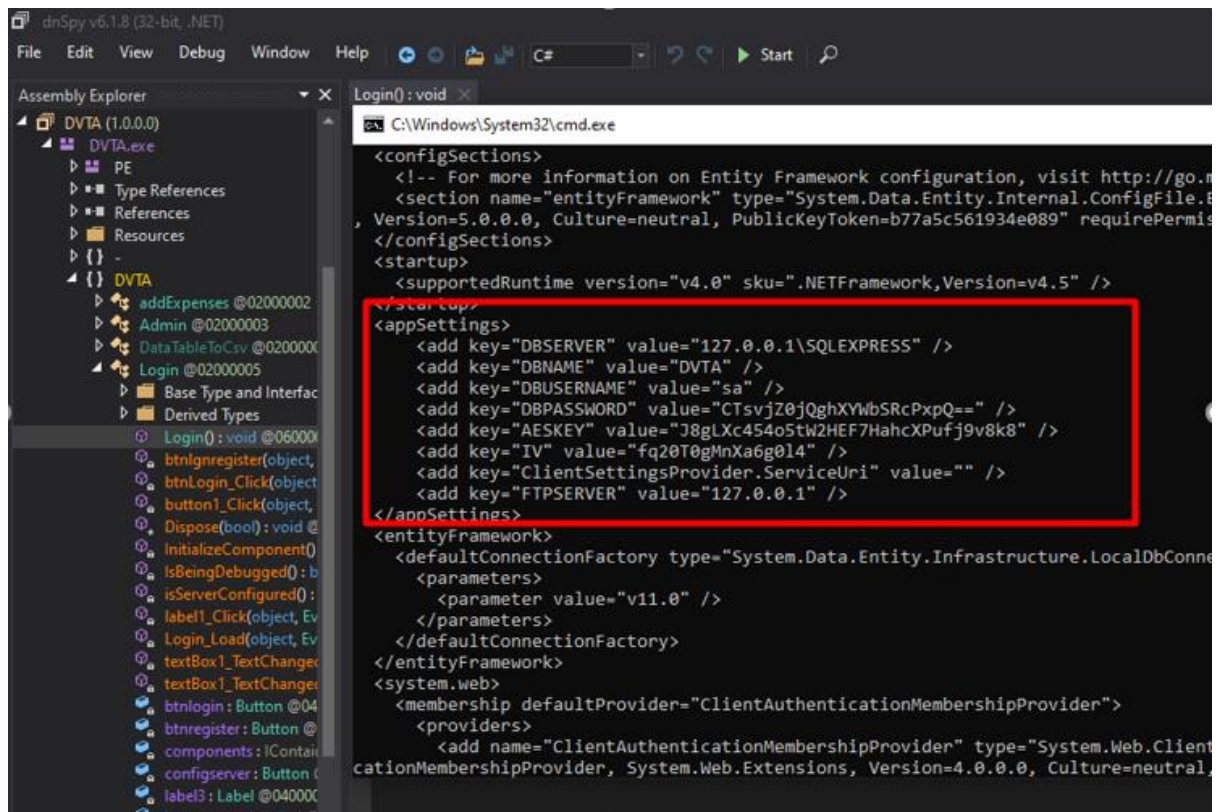


2. Decompile and try to retrieve the source code of the application? Also, check if any hardcoded sensitive information is found?

- By decompiling the application using DNSpy or MS Visual studio tools, we can see the source code of the application.

Assignment 2

CB.SC.P2CYS23017



3) Sniff the traffic between client and server. Identify which protocol is being used for communication?

- With Wireshark we can sniff the client and server
- Next inspect the contents of the packets to determine whether the app is using TCP/UDP protocol for communication.
- In the packet inspection window, we can see that the protocol used by the dvta is TCP protocol.

Assignment 2

CB.SC.P2CYS23017

No.	Time	Source	Destination	Protocol	Length	Info
307	2024/146 22:12:09.278478	2404::6800:4007:813::	2409:4072:6d97:6f54::	TLSv1.3	113	Application Data
308	2024/146 22:12:09.294257	2404::6800:4007:813::	2409:4072:6d97:6f54::	TCP	74	443 → 8166 [ACK] Seq=3882 Ack=3792 Min=76800 Len=0
309	2024/146 22:12:09.298067	2404::6800:4007:813::	2409:4072:6d97:6f54::	TCP	74	443 → 8166 [ACK] Seq=3882 Ack=3877 Min=76800 Len=0
310	2024/146 22:12:09.309943	2404::6800:4007:813::	2409:4072:6d97:6f54::	TLSv1.3	169	Application Data
311	2024/146 22:12:09.309943	2404::6800:4007:813::	2409:4072:6d97:6f54::	TLSv1.3	148	Application Data
312	2024/146 22:12:09.309943	2404::6800:4007:813::	2409:4072:6d97:6f54::	TLSv1.3	113	Application Data
313	2024/146 22:12:09.309985	2409:4072:6d97:6f54::	2404:6800:4007:813::	TCP	74	8166 → 443 [ACK] Seq=3977 Ack=4090 Min=64512 Len=0
314	2024/146 22:12:09.310283	2409:4072:6d97:6f54::	2404:6800:4007:813::	TLSv1.3	113	Application Data
315	2024/146 22:12:09.310500	2409:4072:6d97:6f54::	2404:6800:4007:813::	TLSv1.3	109	Application Data
316	2024/146 22:12:09.324832	2404::6800:4007:813::	2409:4072:6d97:6f54::	TLSv1.3	146	Application Data
317	2024/146 22:12:09.324832	2404::6800:4007:813::	2409:4072:6d97:6f54::	TLSv1.3	148	Application Data
318	2024/146 22:12:09.324832	2404::6800:4007:813::	2409:4072:6d97:6f54::	TLSv1.3	105	Application Data
319	2024/146 22:12:09.324879	2409:4072:6d97:6f54::	2404:6800:4007:813::	TCP	74	8166 → 443 [ACK] Seq=4051 Ack=4267 Min=64512 Len=0
320	2024/146 22:12:09.337689	2404::6800:4007:813::	2409:4072:6d97:6f54::	TLSv1.3	146	Application Data
321	2024/146 22:12:09.337689	2404::6800:4007:813::	2409:4072:6d97:6f54::	TLSv1.3	148	Application Data
322	2024/146 22:12:09.337731	2409:4072:6d97:6f54::	2404:6800:4007:813::	TCP	74	8166 → 443 [ACK] Seq=4051 Ack=4413 Min=64256 Len=0
323	2024/146 22:12:09.342700	2404::6800:4007:813::	2409:4072:6d97:6f54::	TLSv1.3	146	Application Data
324	2024/146 22:12:09.342700	2404::6800:4007:813::	2409:4072:6d97:6f54::	TLSv1.3	148	Application Data
325	2024/146 22:12:09.342700	2404::6800:4007:813::	2409:4072:6d97:6f54::	TLSv1.3	105	Application Data
326	2024/146 22:12:09.342809	2409:4072:6d97:6f54::	2404:6800:4007:813::	TCP	74	8166 → 443 [ACK] Seq=4051 Ack=4590 Min=65536 Len=0
327	2024/146 22:12:09.369233	2404::6800:4007:813::	2409:4072:6d97:6f54::	TCP	74	443 → 8166 [ACK] Seq=4590 Ack=4016 Min=76800 Len=0
328	2024/146 22:12:09.384150	2404::6800:4007:813::	2409:4072:6d97:6f54::	TCP	74	443 → 8166 [ACK] Seq=4590 Ack=4051 Min=76800 Len=0
329	2024/146 22:12:09.500165	TotalLen: 86; 65: 41	Broadcast	ARP	42	Who has 102.168.1.104? Tell 102.168.1.101

<p>> Frame 324: 148 bytes on wire (1184 bits), 148 bytes captured (1184 bits) on interface \Device\NPF_{F...}</p> <p>> Ethernet II, Src: Shenzhen_15:06:92 (7c:45:d0:15:06:92), Dst: IntelCor_86:6f:41 (c0:b8:83:86:6f:41)</p> <p>> Internet Protocol Version 6, Src: 2404:6800:4007:813::2003, Dst: 2409:4072:6d97:6f54::5dec:2035:a5db:4...</p> <p>> Transmission Control Protocol, Src Port: 443, Dst Port: 8166, Seq: 4485, Ack: 3977, Len: 74</p> <p>Source Port: 443</p> <p>Destination Port: 8166</p> <p><Source or Destination Port: 443></p> <p><Source or Destination Port: 8166></p> <p>[Stream index: 0]</p> <p>[Conversation completeness: Incomplete, DATA (15)]</p> <p>[TCP Segment Len: 74]</p> <p>Sequence Number: 4485 (relative sequence number)</p> <p>Sequence Number (raw): 2660817928</p> <p>[Next Sequence Number: 4559 (relative sequence number)]</p> <p>Acknowledgment Number: 3977 (relative ack number)</p>	<pre> 0000 c0 b8 83 86 6f 41 7c 45 d0 15 06 92 86 dd 62 8f oA[Eb 0010 55 72 00 5e 06 78 24 04 68 00 40 07 08 13 00 00 Ur-^xS- h@.... 0020 00 00 00 00 20 03 24 09 40 72 6d 97 6f 54 5d ec 5-@m-oT]... 0030 2d 35 a5 db 4c af 01 bb 1f e6 9e 98 dc 08 cb 66 ...S-L.....f 0040 ab d1 50 18 01 2c 56 e0 00 00 17 03 03 00 45 aa ...P-,V.....E- 0050 c6 03 6a 68 ba 24 9b 1f 6c ee 27 6d b3 60 dd 81 ...jh\$- 1'm"-... 0060 6d ef 2b 0c fe c9 1d 2f 61 84 f9 9b 64 d4 63 fa m+.../ a-o-d-c- 0070 ce 1a e7 ce 6b b9 11 ec 25 e6 b5 91 2e ec 15 98 1--X-.... 0080 ad b1 f7 b2 5d 3a 2b 16 a9 44 d7 af 13 d1 0d 52 ...[]+-D.....R 0090 fd cc 99 21 ...] </pre>
--	--

4) Identify if unencrypted communication is happening between client and server?

- In this case we can use either ECHIMIRAGE / wireshark. We have used Echomirage here.
- From the output we can see that when we login to DVTA , the data is sent as plaintext format to the database.

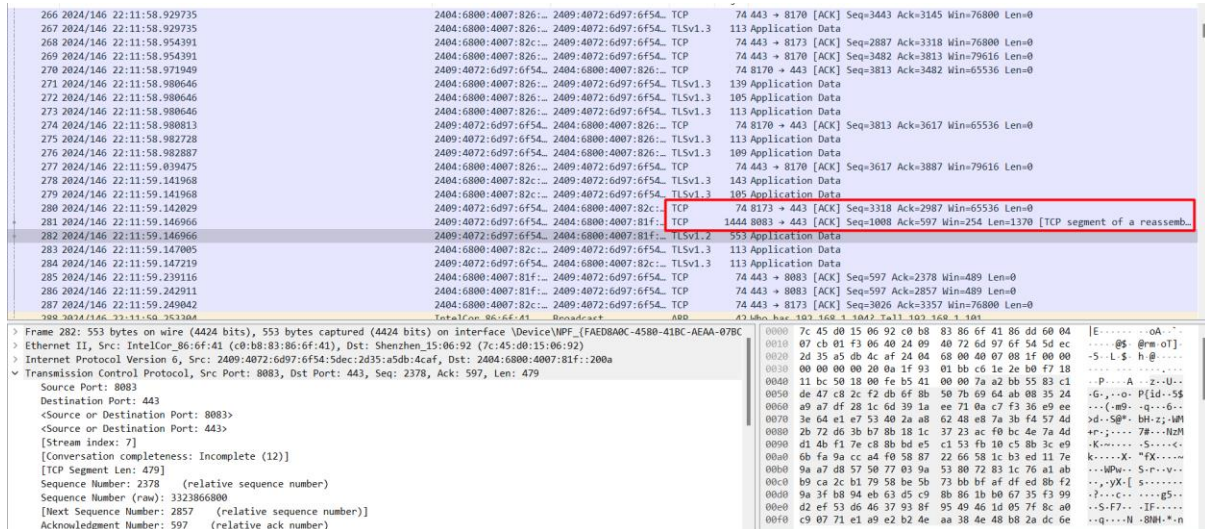
Traffic Log	Rules	Intercept
Outbound TCP data to 192.168.56.110:1433		
0x0000	01 09 00 B2 00 00 01 00 16 00 00 00 12 00 00 00*
0x0010	02 00 00 00 00 00 00 00 00 00 01 00 00 00 73 00s.
0x0020	65 00 6C 00 65 00 63 00 74 00 20 00 69 00 74 00	e.l.e.c.t..i.t.
0x0030	65 00 6D 00 2C 00 20 00 70 00 72 00 69 00 63 00	e.m.,.p.r.i.c.
0x0040	65 00 2C 00 20 00 64 00 61 00 74 00 65 00 2C 00	e.,.d.a.t.e.,.
0x0050	74 00 69 00 6D 00 65 00 20 00 66 00 72 00 6F 00	t.i.m.e..f.r.o.
0x0060	6D 00 20 00 65 00 78 00 70 00 65 00 6E 00 73 00	m..e.x.p.e.n.s.
0x0070	65 00 73 00 20 00 77 00 68 00 65 00 72 00 65 00	e.s..w.h.e.r.e.
0x0080	20 00 65 00 6D 00 61 00 69 00 6C 00 3D 00 27 00	.e.m.a.i.l.=.'
0x0090	72 00 61 00 79 00 6D 00 6F 00 6E 00 64 00 40 00	r.a.y.m.o.n.d. @.
0x00A0	74 00 65 00 73 00 74 00 2E 00 63 00 6F 00 6D 00	t.e.s.t...c.o.m.
0x00B0	27 00	'.

5) Capture and analyse the communication using proxy tools (eg: Burpsuite, Echo mirage).

Assignment 2

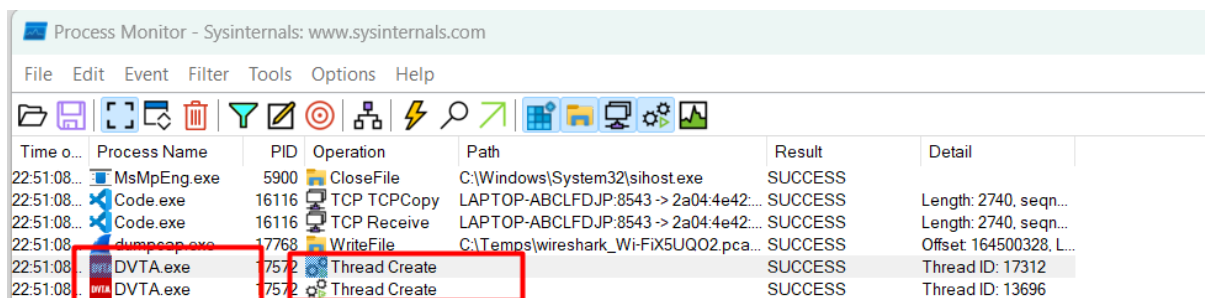
CB.SC.P2CYS23017

- From the below screenshot, we can understand that using Wireshark we're able to capture & analyse the requests that are being sent to the database and to the server.



6) Analyse the application workflow and observe which all files/folders are being used by the application using Process Monitor

- With the help of a tool called Process-Monitor can see that there are several files & folders being retrieved when running the DVTA.exe.



7) Exploit DLL Hijacking vulnerability (You can use a simple legitimate "Hello World" printing dll.

- In order to hijack a DLL, we need to find which DLL's that are being loaded when DVTA.exe runs is not found.
- For this we need to open Procmon & set the following 3 filters .

Assignment 2

CB.SC.P2CYS23017

Name	Date modified	Type	Size
DBAccess.dll	03-09-2023 14:49	Application extens...	8 KB
DBAccess.pdb	03-09-2023 14:49	PDB File	16 KB
DVTA.exe	03-09-2023 14:49	Application	218 KB
DVTA.exe.Config	03-09-2023 14:50	Configuration Sou...	3 KB
DVTA.pdb	03-09-2023 14:49	PDB File	54 KB
DVTA.vshost.exe	03-09-2023 14:49	Application	23 KB
DVTA.vshost.exe.config	03-09-2023 14:49	Configuration Sou...	2 KB
DVTA.vshost.exe.manifest	03-09-2023 14:49	MANIFEST File	1 KB
EntityFramework.dll	03-09-2023 14:49	Application extens...	1,091 KB
EntityFramework.xml	03-09-2023 14:49	XML File	1,094 KB
ExcelLibrary.dll	03-09-2023 14:49	Application extens...	116 KB
hello-world-x64.dll	25-05-2024 22:54	Application extens...	11 KB

- We will Start Process Monitor Filter

Process Monitor Filter

Display entries matching these conditions:

Path contains dll then Include

Reset Add Remove

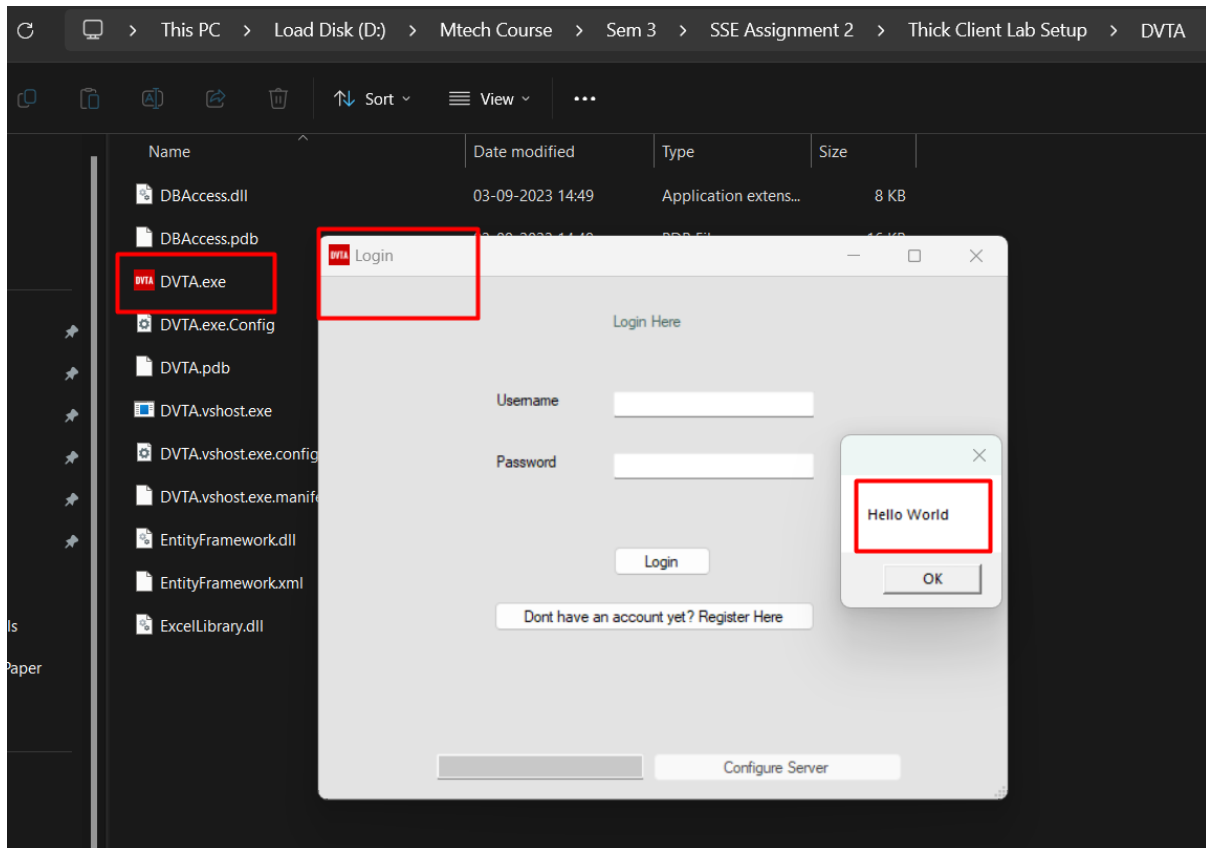
Column	Relation	Value	Action
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Process N...	contains	dvta	Include
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Result	is	NAME NOT FOU...	Include
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Path	contains	dll	Include
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Process N...	is	Procmon.exe	Exclude
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Process N...	is	Procexp.exe	Exclude
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Process N...	is	Autoruns.exe	Exclude
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Process N...	is	Procmon64.exe	Exclude
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Process N...	is	Procexp64.exe	Exclude
<input checked="" type="checkbox"/> <input checked="" type="checkbox"/> Process N...	is	System	Exclude

OK Cancel Apply

Assignment 2

CB.SC.P2CYS23017

- When click DVTA.exe automatically Hello world pop up will appear with opening of DVTA Login Page



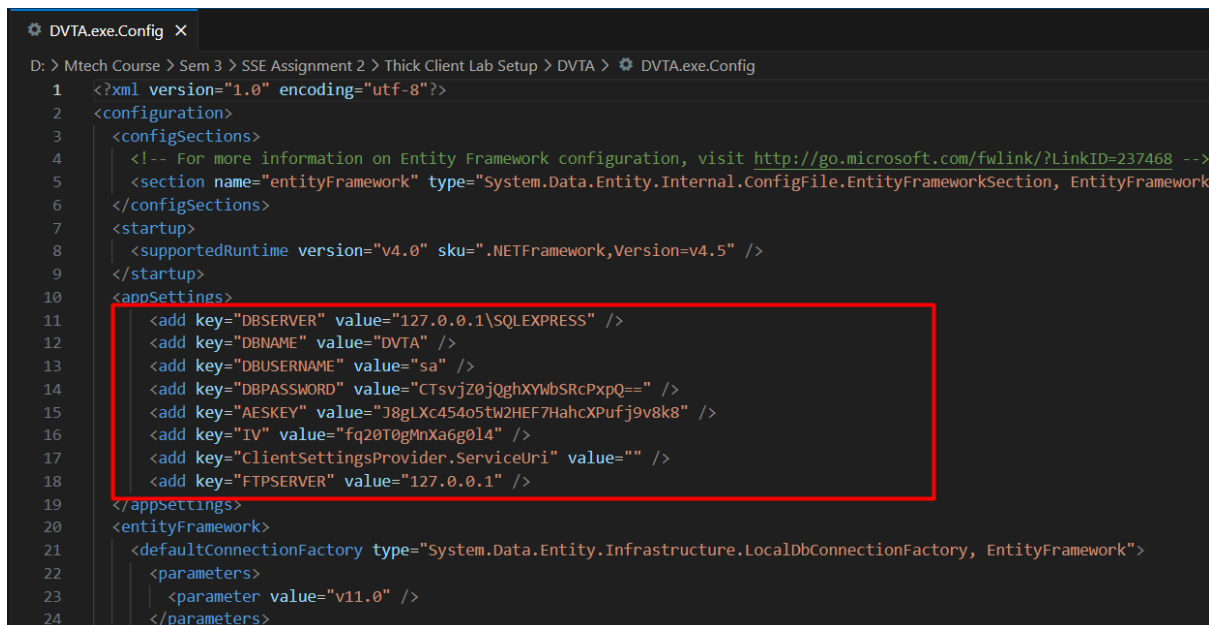
- As we can see now when the DVTA.exe runs, it loads our calc.dll along with the application. Thus we have hijacked the DLL.

8) Check for sensitive information in the configuration files of the thick client application?

- In the folder of DVTA, we have few files . One of the files is App.config. It contains the following sensitive information.
- We have to open Visual Studio and analyse DVTA.exe.config.

Assignment 2

CB.SC.P2CYS23017



```

1  <?xml version="1.0" encoding="utf-8"?>
2  <configuration>
3    <configSections>
4      <!-- For more information on Entity Framework configuration, visit http://go.microsoft.com/fwlink/?LinkID=237468 -->
5      <section name="entityFramework" type="System.Data.Entity.Internal.ConfigFile.EntityFrameworkSection, EntityFramework" />
6    </configSections>
7    <startup>
8      <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.5" />
9    </startup>
10   <appSettings>
11     <add key="DBSERVER" value="127.0.0.1\SQLEXPRESS" />
12     <add key="DBNAME" value="DVTA" />
13     <add key="DBUSERNAME" value="sa" />
14     <add key="DBPASSWORD" value="CTsvjZ0jQghXYWbSRcPxpQ==" />
15     <add key="AESKEY" value="J8gLXc454o5tW2HEF7HahcXPufj9v8k8" />
16     <add key="IV" value="fq20T0gMnXa6g014" />
17     <add key="ClientSettingsProvider.ServiceUri" value="" />
18     <add key="FTPSERVER" value="127.0.0.1" />
19   </appSettings>
20   <entityFramework>
21     <defaultConnectionFactory type="System.Data.Entity.Infrastructure.LocalDbConnectionFactory, EntityFramework">
22       <parameters>
23         <parameter value="v11.0" />
24       </parameters>

```

9) Identify sensitive information found in memory?

- From the source code which we got from DNSpy, we got to know that it stores the username & password in HKCU/dvta registry file.
- We can visit the registry to find the sensitive information which is stored in the memory.
- We have to open registry editor to analyse dvta username and password.

