

Thick Client Penetration Testing: Information Gathering

February 2, 2021 By Raj Chandel

In the previous article, we have discussed the reverse engineering of original DVTA application in the [Lab setup of Thick Client: DVTA part 2](#)

In this part, we are going to systematically pentesting the DVTA application for various issues.

Table of Content

- Prerequisites
- Information Gathering by using CFF Explorer
- Information Gathering by using Sysinternal Suite
- Information Gathering by using Procmon
- Information Gathering by using Wireshark

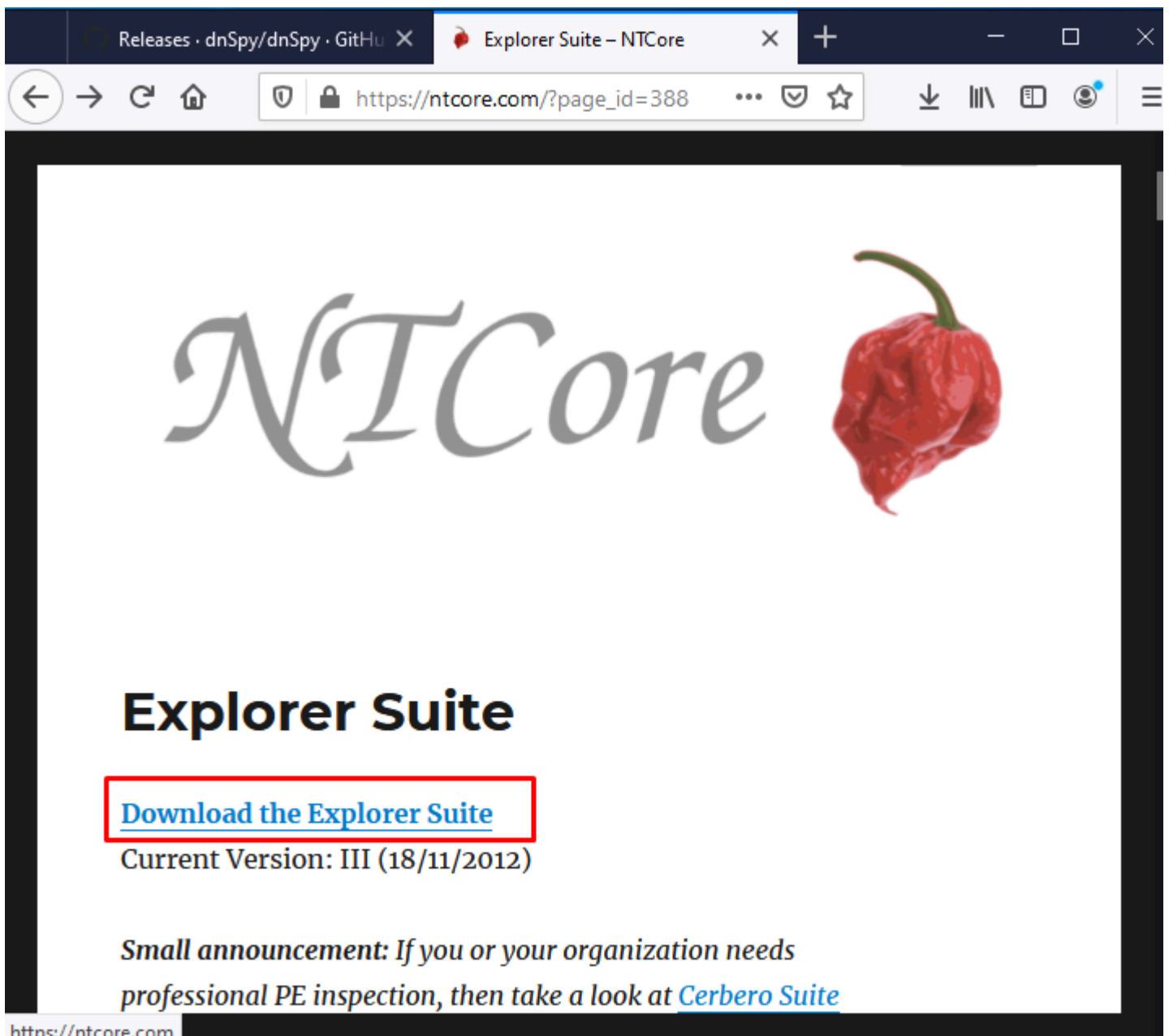
Prerequisites

- CFF explorer
- Sysinternals Suite
- Wireshark

Information gathering by using CFF Explorer

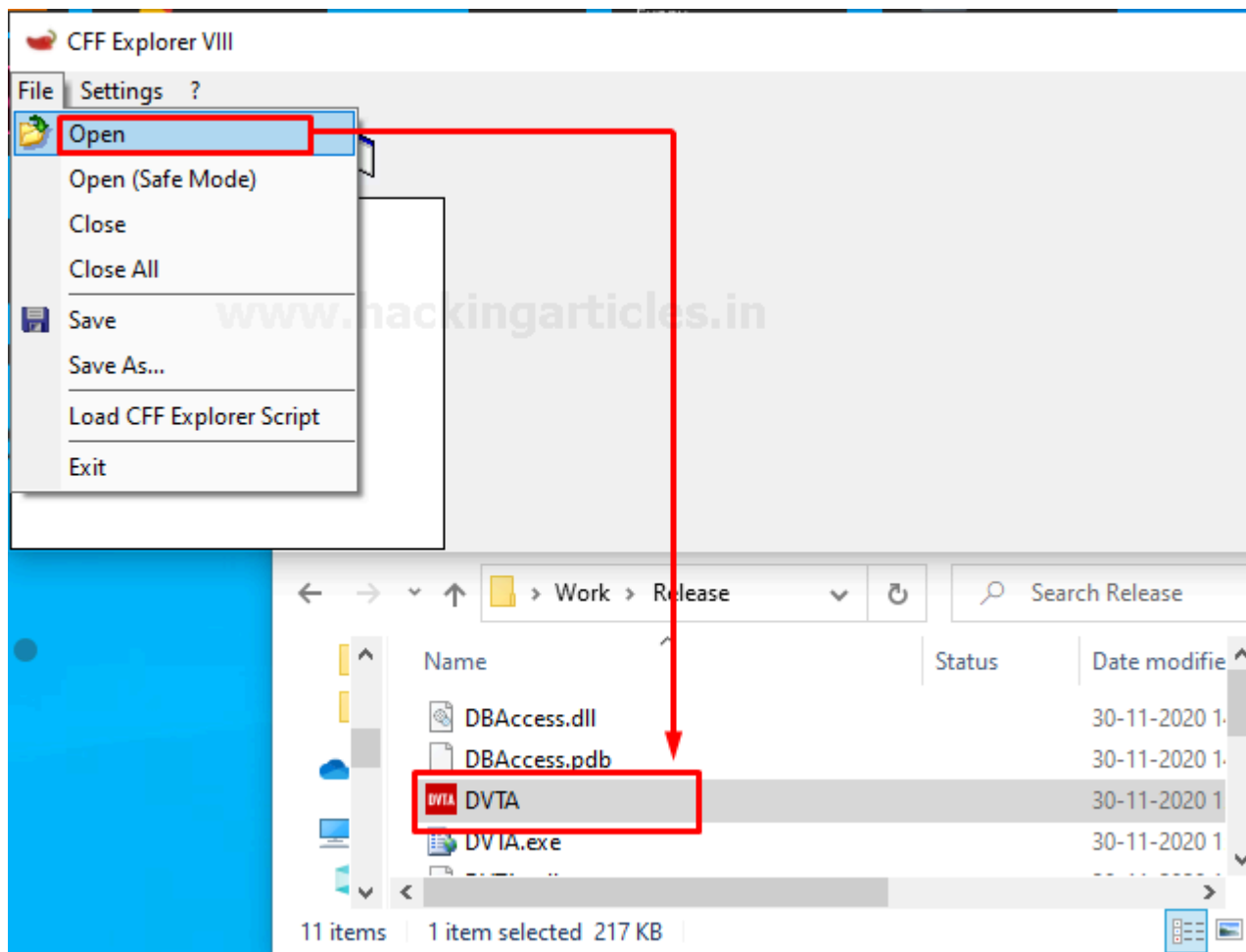
You can directly install CFF Explorer from here: – <https://ntcore.com/files/ExplorerSuite.exe>

CFF Explorer was designed to make PE editing as easy as possible, but without losing sight on the portable executable's internal structure. This application includes a series of tools which might help not only reverse engineers but also programmers.



Let's open up the CFF explorer. Basically, CFF explorer is used to checking how the binary is built. Having knowledge on what languages are used for building the binary is really useful because if it is built using Java, C, C++, ML, AI... we can look for vulnerabilities like Buffer overflow if it is built using C-sharp.net kind of languages it is very easy to decompile the application and reverse engineered

So, without wasting of much time quickly open up the DVTA.exe by navigating to **File > open > DVTA.exe** as shown in below image.



Let's see which type of juicy information we get ...

So if you notice by 2nd line this is a Portable Executable 32.NET Assembly and Microsoft Visual .NET is used to build this. So, as you can see this binary is built using Microsoft Visual Studio .NET and it's a .NET assembly so that we can de-compile this application very easily.

CFF Explorer VIII - [DVTa.exe]

File Settings ?

File: DVTa.exe

- Dos Header
- Nt Headers
 - File Header
 - Optional Header
 - Data Directories [x]
- Section Headers [x]
- Import Directory
- Resource Directory
- Relocation Directory
- Debug Directory
- .NET Directory
 - MetaData Header
 - MetaData Streams
 - #~
 - Tables Header
 - Tables
 - #Strings
 - #US
 - #GUID
 - #Blob

Address Converter

Dependency Walker

Hex Editor

Identifier

Import Adder

Quick Disassembler

Rebuilder

Resource Editor

DVTa.exe

Property	Value
File Name	C:\Users\vijvi\OneDrive\Desktop\Work\Release\DVTa.exe
File Type	Portable Executable 32 .NET Assembly
File Info	Microsoft Visual Studio .NET
File Size	217.00 KB (222208 bytes)
PE Size	217.00 KB (222208 bytes)
Created	Friday 17 January 2020, 22.10.00
Modified	Monday 30 November 2020, 15.29.32
Accessed	Tuesday 19 January 2021, 09.50.08
MD5	C187E83253AB4F35E8F16CF6A821F34F
SHA-1	7F4A87CB4334E33E6EEB483C40241A028ADCC655

Property	Value
Comments	
CompanyName	Microsoft
FileDescription	DVTa
FileVersion	1.0.0.0
InternalName	DVTa.exe
LegalCopyright	Copyright © Microsoft 2016
LegalTrademarks	
OriginalFilename	DVTa.exe
ProductName	DVTa
ProductVersion	1.0.0.0

This information is enough to decompile the application easily.

Information Gathering by using Sysinternals Suite

Download and install Sysinternal Suite: <https://download.sysinternals.com/files/SysinternalsSuite.zip>

Once the download complete, install it. Installation is pretty much easy there is no need to do changes in installation just simply press next and next until installation doesn't finish.

Releases · d Explorer Su Wireshark Sysinter X

← → ↺ 🏠 🔒 https://docs.microsoft.com/en-us/ ⌵ ⋮ ⬇ 📖 👤 ⋮

Microsoft Documentation

Docs / Sysinternals / Downloads

☰ Table of contents ⋮

Sysinternals Suite

11/04/2020 • 2 minutes to read • 🌱 📄 👤 🌱 👤 +2

www.hackingarticles.in

In this article

[Introduction](#)

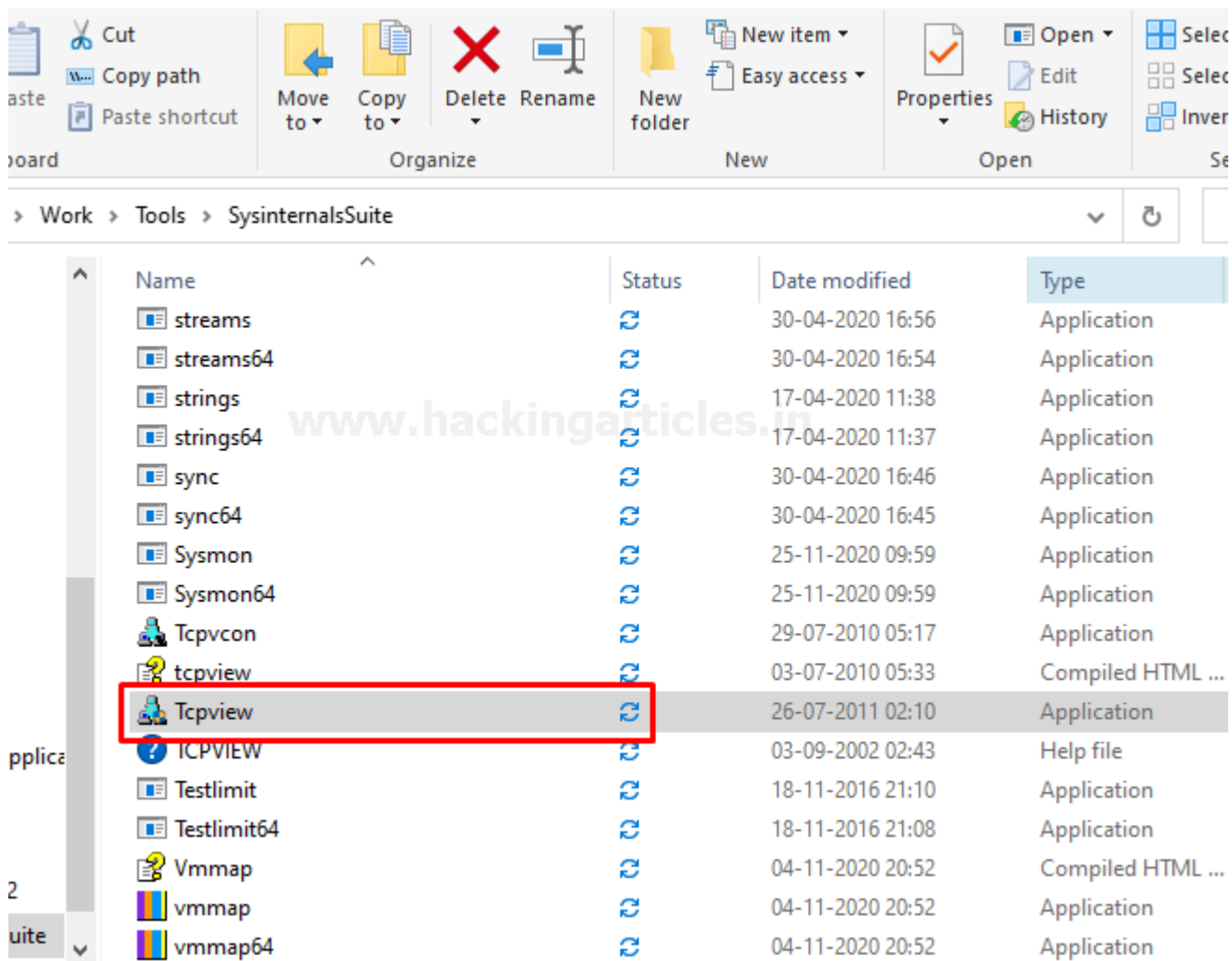
By Mark Russinovich
Updated: November 04, 2020

[Download Sysinternals Suite \(38 MB\)](#) ←

[Download Sysinternals Suite for Nano Server \(7.8 MB\)](#)

[Download Sysinternals Suite for ARM64 \(9.6 MB\)](#)

Let's begin with a tool TCP View from the Sysinternals Suite



Open up the application TCP view and just after opening the application you can see the machine is making network communications and what we're interested in is DVTA application.

So, the thing is we have to gather information like where these applications communicating such as the Destination IP address.

TCPView - Sysinternals: www.sysinternals.com

File Options Process View Help

Process	PID	Protocol	Local Address	Local Port	Remote Address	Remote Port	State
FileZilla Serve...	5060	TCPV6	desktop-km8252d	ftp	desktop-km8252d	0	LISTENING
FileZilla Serve...	5060	TCPV6	[0:0:0:0:0:0:1]	14147	desktop-km8252d	0	LISTENING
firefox.exe	5840	TCP	DESKTOP-KM825...	49850	localhost	49851	ESTABLISHED
firefox.exe	5840	TCP	DESKTOP-KM825...	49851	localhost	49850	ESTABLISHED
firefox.exe	1792	TCP	DESKTOP-KM825...	49852	localhost	49853	ESTABLISHED
firefox.exe	1792	TCP	DESKTOP-KM825...	49853	localhost	49852	ESTABLISHED
firefox.exe	7400	TCP	DESKTOP-KM825...	49854	localhost	49855	ESTABLISHED
firefox.exe	7400	TCP	DESKTOP-KM825...	49855	localhost	49854	ESTABLISHED
firefox.exe	7332	TCP	DESKTOP-KM825...	49858	localhost	49859	ESTABLISHED
firefox.exe	7332	TCP	DESKTOP-KM825...	49859	localhost	49858	ESTABLISHED
firefox.exe	12060	TCP	DESKTOP-KM825...	50269	localhost	50270	ESTABLISHED
firefox.exe	12060	TCP	DESKTOP-KM825...	50270	localhost	50269	ESTABLISHED
firefox.exe	12448	TCP	DESKTOP-KM825...	61296	localhost	61297	ESTABLISHED
firefox.exe	12448	TCP	DESKTOP-KM825...	61297	localhost	61296	ESTABLISHED
firefox.exe	5840	TCP	desktop-km8252d	61360	ec2-54-149-138-1...	https	ESTABLISHED
firefox.exe	11640	TCP	DESKTOP-KM825...	61362	localhost	61363	ESTABLISHED
firefox.exe	11640	TCP	DESKTOP-KM825...	61363	localhost	61362	ESTABLISHED
firefox.exe	12560	TCP	DESKTOP-KM825...	61420	localhost	61421	ESTABLISHED
firefox.exe	12560	TCP	DESKTOP-KM825...	61421	localhost	61420	ESTABLISHED
lsass.exe	808	TCP	DESKTOP-KM825...	49664	DESKTOP-KM825...	0	LISTENING
lsass.exe	808	TCPV6	desktop-km8252d	49664	desktop-km8252d	0	LISTENING
OfficeClickTo...	4092	TCP	desktop-km8252d	61428	52.109.8.19	https	ESTABLISHED
services.exe	796	TCP	DESKTOP-KM825...	49678	DESKTOP-KM825...	0	LISTENING
services.exe	796	TCPV6	desktop-km8252d	49678	desktop-km8252d	0	LISTENING
spoolsv.exe	3752	TCP	DESKTOP-KM825...	49668	DESKTOP-KM825...	0	LISTENING

Endpoints: 93 Established: 19 Listening: 35 Time Wait: 1 Close Wait: 0

To do this launch the modified DVTA application and see what happens... 🤔

First of all, configure the server to your local Ip address or the loopback address and the login to the DVTA application with the user credentials.

DVTA Login

Login Here

Username

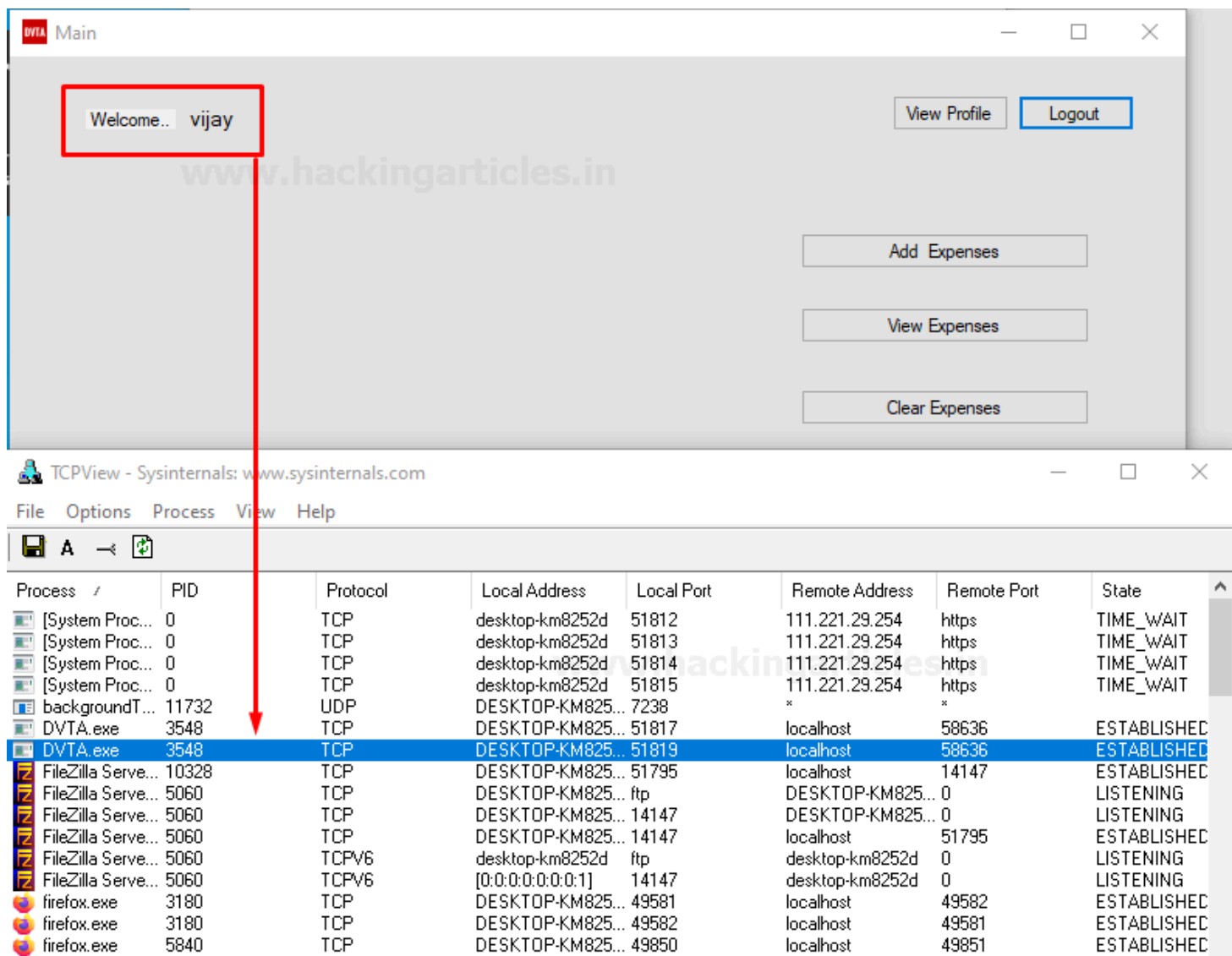
Password

Login

Dont have an account yet? Register Here

Configure Server

Now, you can see there is a new entry in TCP view showing DVTA.exe processes and the local address is 127.0.0.1 and the remote address that is communicating with localhost in this case because it's the loopback address where the database is installed.



In this way, we can figure out the IP addresses that our applications are communicating with. So, we have completed the first step towards the information gathering.

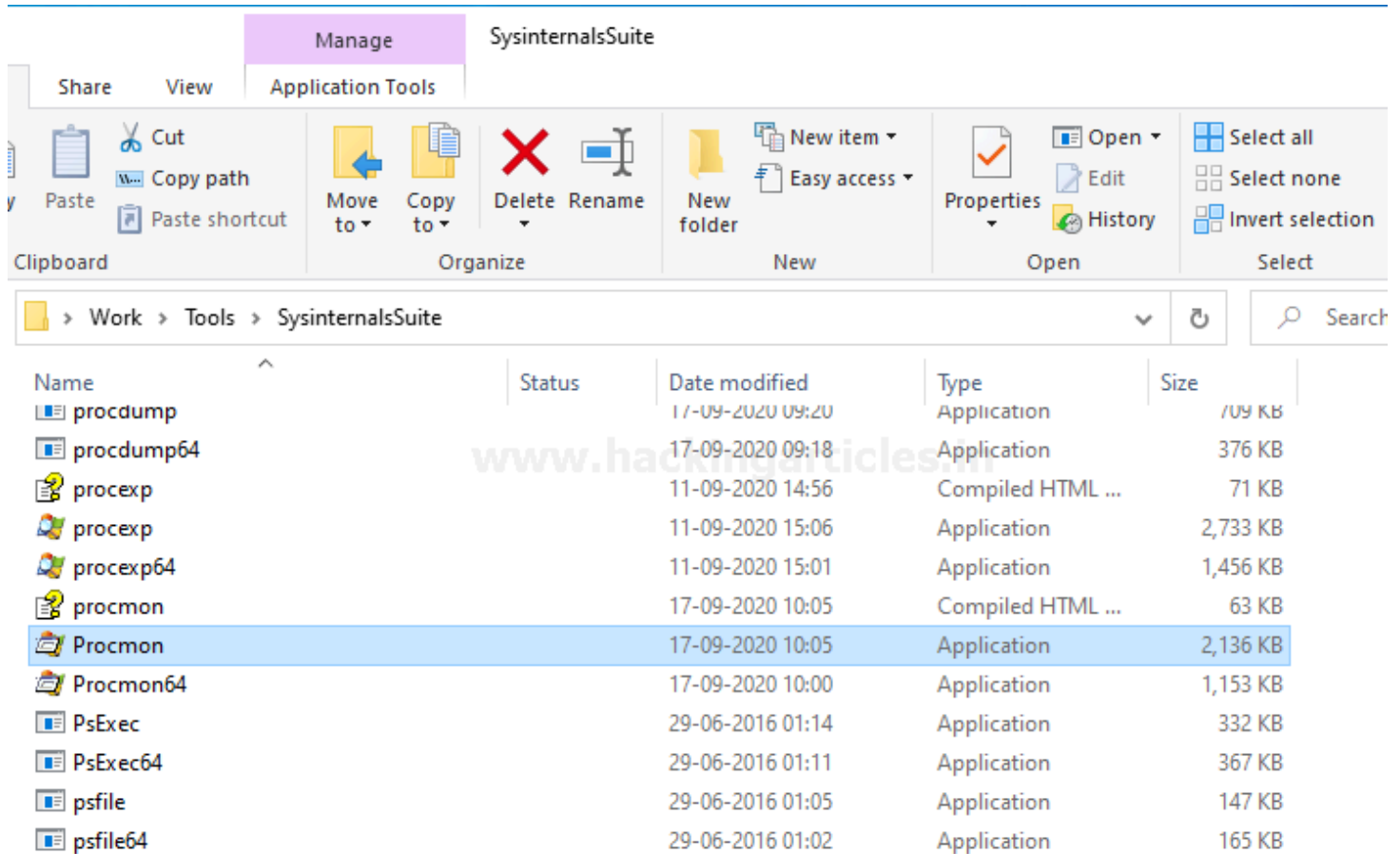
Let's move to another tool from the Sysinternals Suite called Procmon.

Information Gathering by using Procmon

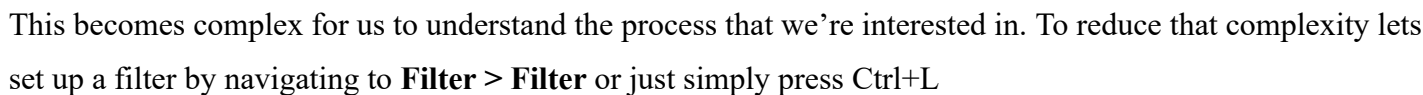
Basically, Procmon is a process monitor that helps us to understand the file system changes and what is being accessed on the file system.

Let's open up the application Procmon which you can find in the folder of Sysinternals suite.

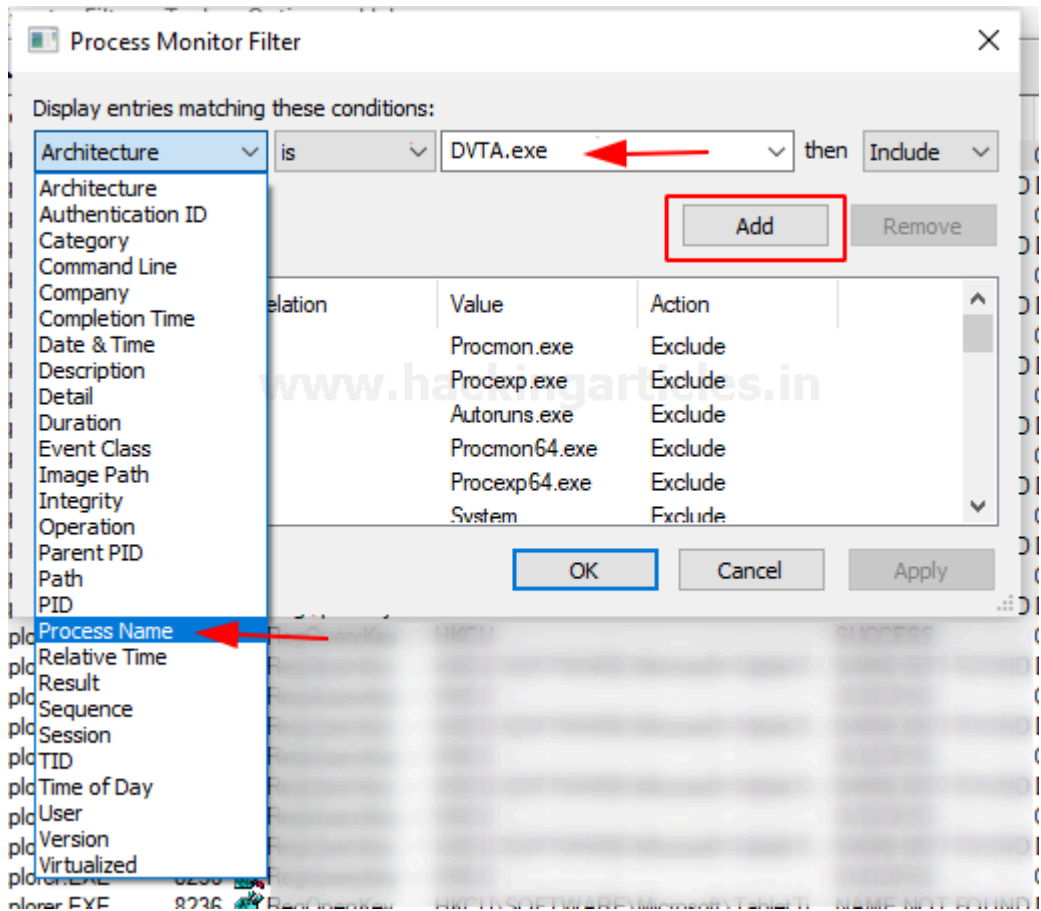
Here we're using 32-bit of version.



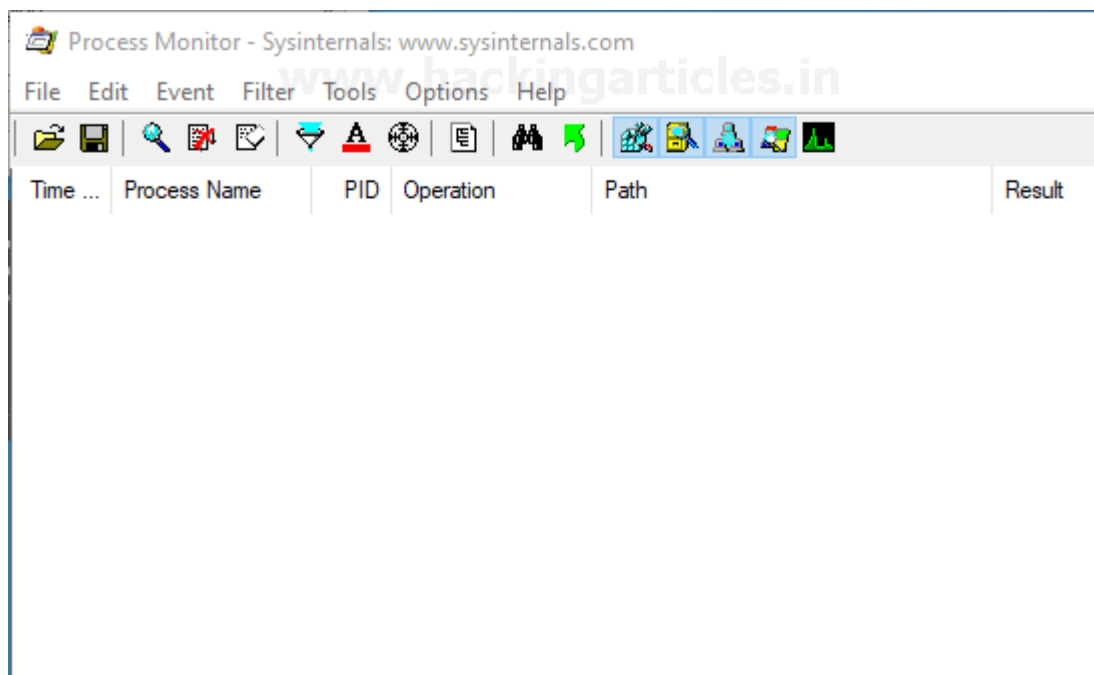
Just after the opening of Procmon application, you can see this process monitor is showing lot's of traffic and the reason behind of that is it picks up all of the processes and shows the output of all the processes In its window.



Select the Attribute to **Process Name** is **DVTA.exe** because this is the only process that what we're interested in and **ADD** the filter.

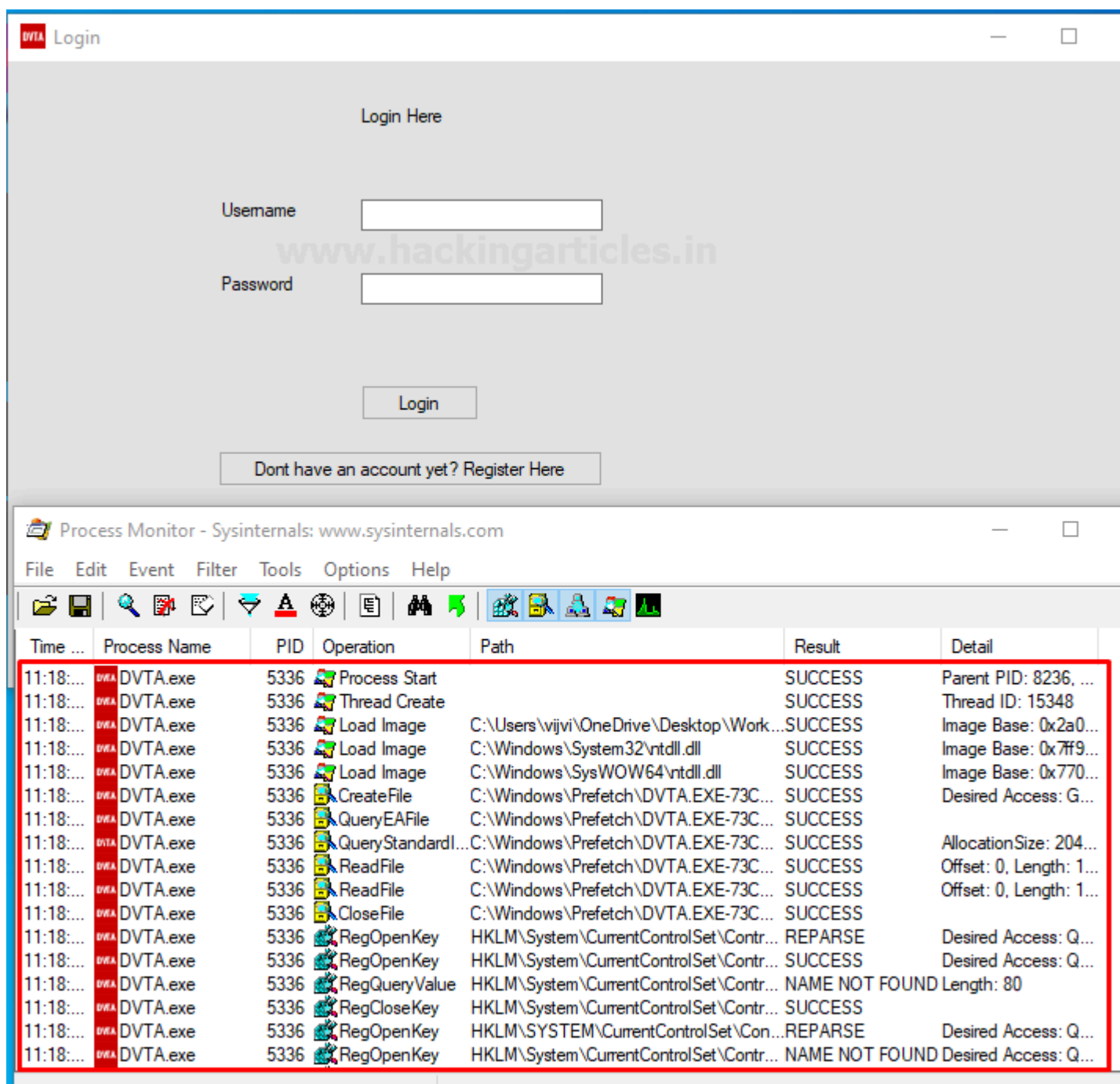


Now you are going to see the screen empty because DVTA is not currently running.



So, without wasting of time go back to modified DVTA application folder and run the DVTA.exe Now just by the opening of DVTA, there is traffic generated only for the DVTA.exe file. So now they become more

interesting and easier to understand.



Now configure our server to 127.0.0.1. we have configured the server already in previous articles we're doing this again to understand what changes are being made in the file system when we configure it. But before doing this clear your screen to reduce the complexity by navigating to **Edit > Clear Display** so that when we press configure server button, we can only see which changes being made.

Just after hitting the Configure server button, you can see there is instantly lots of traffic generated as shown below In the image.

DVIA Login

Login Here

Username

Password

[www.hackingarticles.in](#)

Process Monitor - Sysinternals: www.sysinternals.com

File Edit Event Filter Tools Options Help

Time ...	Process Name	PID	Operation	Path	Result
11:32:...	DVTA.exe	7760	CreateFile	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	QueryNetwork...	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	CloseFile	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	CreateFile	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	QueryNetwork...	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	CloseFile	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	CreateFile	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	QueryStandardI...	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	ReadFile	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	ReadFile	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	ReadFile	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	ReadFile	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	ReadFile	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	ReadFile	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	ReadFile	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	ReadFile	C:\Windows\Microsoft.NET\Framework...	SUCCESS
11:32:...	DVTA.exe	7760	ReadFile	C:\Windows\Microsoft.NET\Framework...	SUCCESS

Showing 1061 of 366882 events (0.28%) Backed by virtual memory

Now, what we interested in we want to understand what happens when we clicked on configure server button. So just scroll down to the output and see what we can find anything interesting.

There are lots of registry entry being accessed and at the bottom, there are some files that are being accessed such as create the file, Query Network open information file, Close File ...

Process Monitor - Sysinternals: www.sysinternals.com

File Edit Event Filter Tools Options Help

Time ...	Process Name	PID	Operation	Path	Result
11:34:...	DVTA.exe	7760	QueryStandardInformationFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	WriteFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CloseFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CreateFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	QueryNetworkOpenInformationFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CloseFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CreateFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CloseFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CreateFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	QueryNetworkOpenInformationFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CloseFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CreateFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	SetBasicInformationFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CloseFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CreateFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	QueryNetworkOpenInformationFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CloseFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CreateFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	QueryBasicInformationFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS

Showing 211 of 1056845 events (0.019%) Backed by virtual memory

so, it looks like these files are something which is being accessed by DVTA.exe when we try configuring the IP address. So quickly open it up and see what it has. We can directly jump from process monitor to the location of this file by giving it a **right-click** and then navigate to **Jump TO...**

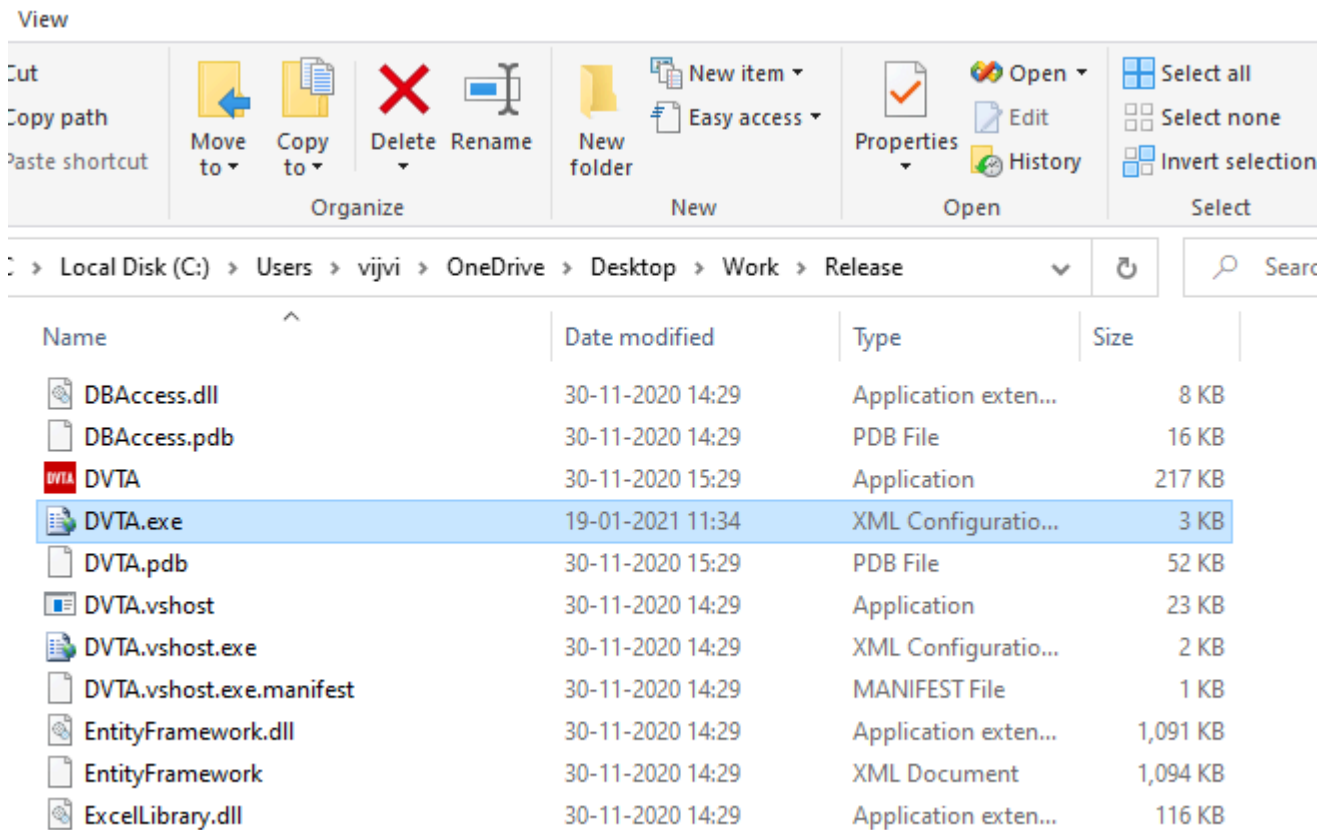
Process Monitor - Sysinternals: www.sysinternals.com

File Edit Event Filter Tools Options Help

Time ...	Process Name	PID	Operation	Path	Result
11:34:...	DVTA.exe	7760	QueryStandardInformationFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	WriteFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CloseFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CreateFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	QueryNetworkOpenInformationFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CloseFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CreateFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CloseFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CreateFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	QueryNetworkOpenInformationFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CloseFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CreateFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	SetBasicInformationFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CloseFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CreateFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	QueryNetworkOpenInformationFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CloseFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	CreateFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS
11:34:...	DVTA.exe	7760	QueryBasicInformationFile	C:\Users\vijvi\OneDrive\Desktop\Work...	SUCCESS

Properties... Ctrl+P
Stack... Ctrl+K
Toggle Bookmark Ctrl+B
Jump To... Ctrl+J
Search Online...
Include 'CloseFile'
Exclude 'CloseFile'
Highlight 'CloseFile'
Copy 'CloseFile'

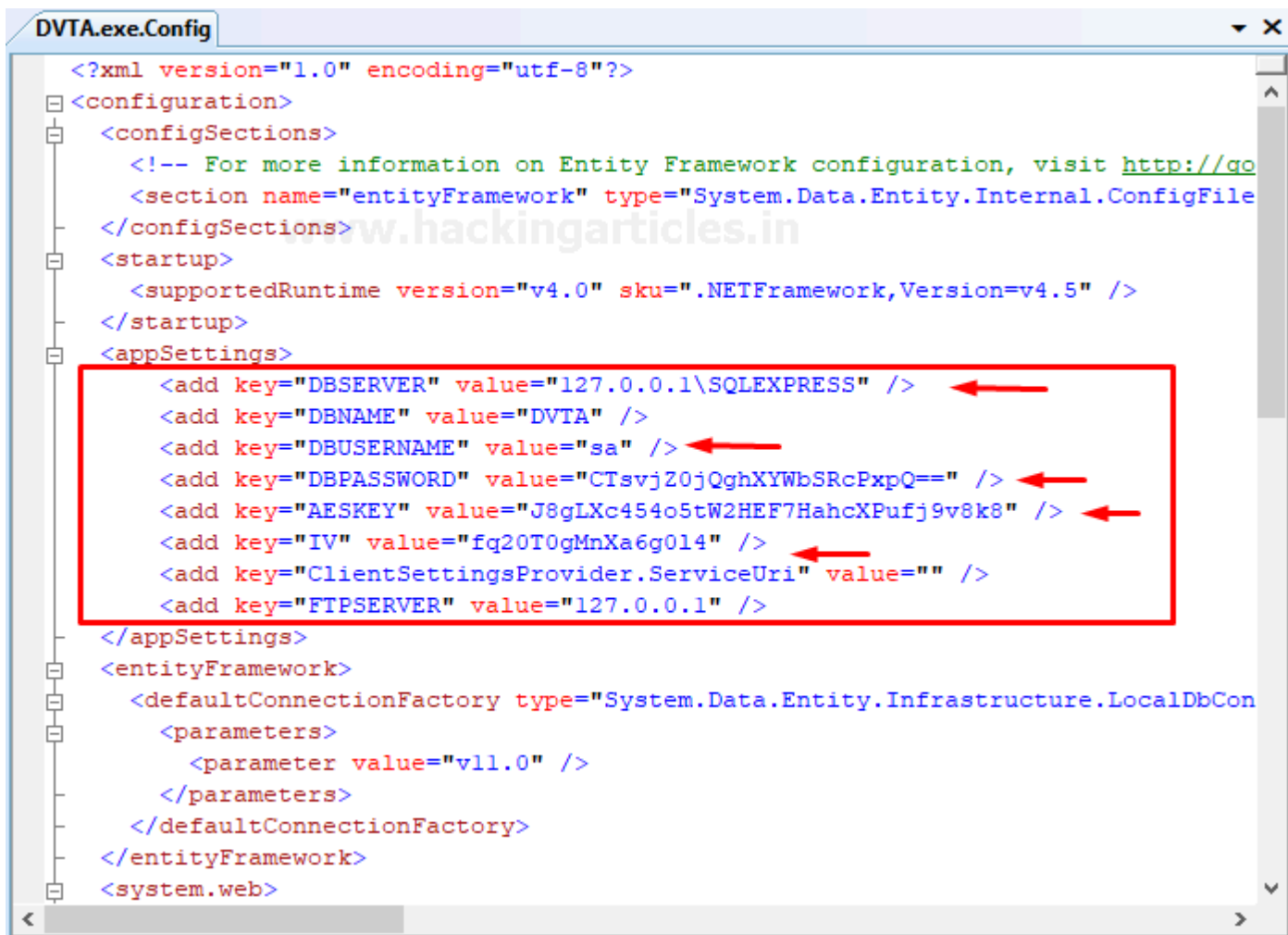
It will redirect you the location of that file changed in the file system are being made as shown below also it automatically highlights to the file where changes are being implied.



Now let's open up the DVTa.exe XML configuration file in you favourite XML viewer.

It's gonna very interesting as you can see the server is configured to **127.0.0.1** exactly the same IP address that we have entered earlier and looks when we click on configure the server the Ip address is taken from the text box and placed here... 🤔

Apart from these entries here we have some other interesting entries if you notice there is a database name which is **DVTA** ... if you remember this is the name of the database we have used when creating a database in SQL Server Express. Also, there is the user name and the password for the database but it seems like the password is Encrypted but if you observe just right below that there is an AES encryption key possibly used for decrypting the Encrypted value and apart from that there is **IV value which is probably used with the AES Encryption**.



```
<?xml version="1.0" encoding="utf-8"?>
<configuration>
  <configSections>
    <!-- For more information on Entity Framework configuration, visit http://go
    <section name="entityFramework" type="System.Data.Entity.Internal.ConfigFile
  </configSections>
  <startup>
    <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.5" />
  </startup>
  <appSettings>
    <add key="DBSERVER" value="127.0.0.1\SQLEXPRESS" />
    <add key="DBNAME" value="DVTA" />
    <add key="DBUSERNAME" value="sa" />
    <add key="DBPASSWORD" value="CTsvjZ0jQghXYWbSRcPxpQ==" />
    <add key="AESKEY" value="J8gLXc454o5tW2HEF7HahcXPufj9v8k8" />
    <add key="IV" value="fq20T0gMnXa6g014" />
    <add key="ClientSettingsProvider.ServiceUri" value="" />
    <add key="FTPSEVER" value="127.0.0.1" />
  </appSettings>
  <entityFramework>
    <defaultConnectionFactory type="System.Data.Entity.Infrastructure.LocalDbCon
    <parameters>
      <parameter value="v11.0" />
    </parameters>
    </defaultConnectionFactory>
  </entityFramework>
  <system.web>
```

Now the main thing to be noted is that when we tried configuring the server in the application... the application is accessing this DVTA.exe.config file and it is trying to place the IP address place there then we have also noticed some more interesting information such as Encrypted Database password and the AES key.

Let's open up the DVTA application and login to the application by using user credentials as shown below and on the other tab process monitor start capturing the traffic.

The image shows a Windows desktop with two applications. The top application is 'DVTA Login', which has a login form with fields for 'Username' (containing 'vijay') and 'Password' (masked with dots). A 'Login' button is highlighted with a red box, and a red arrow points from it to the Process Monitor application below. The Process Monitor application shows a list of registry operations performed by 'DVTA.exe' (PID 6004). The operations include 'RegOpenKey', 'RegQueryKey', 'RegQueryValue', and 'RegOpenKey' on various registry paths, including 'HKU\DEFAULT', 'HKU\DEFAULT\Software\Microsoft', and 'HKLM\Software\Wow6432N...'. The list is also highlighted with a red box.

DVTA Login

Login Here

Username: vijay

Password:

Login

Dont have an account yet? Register Here

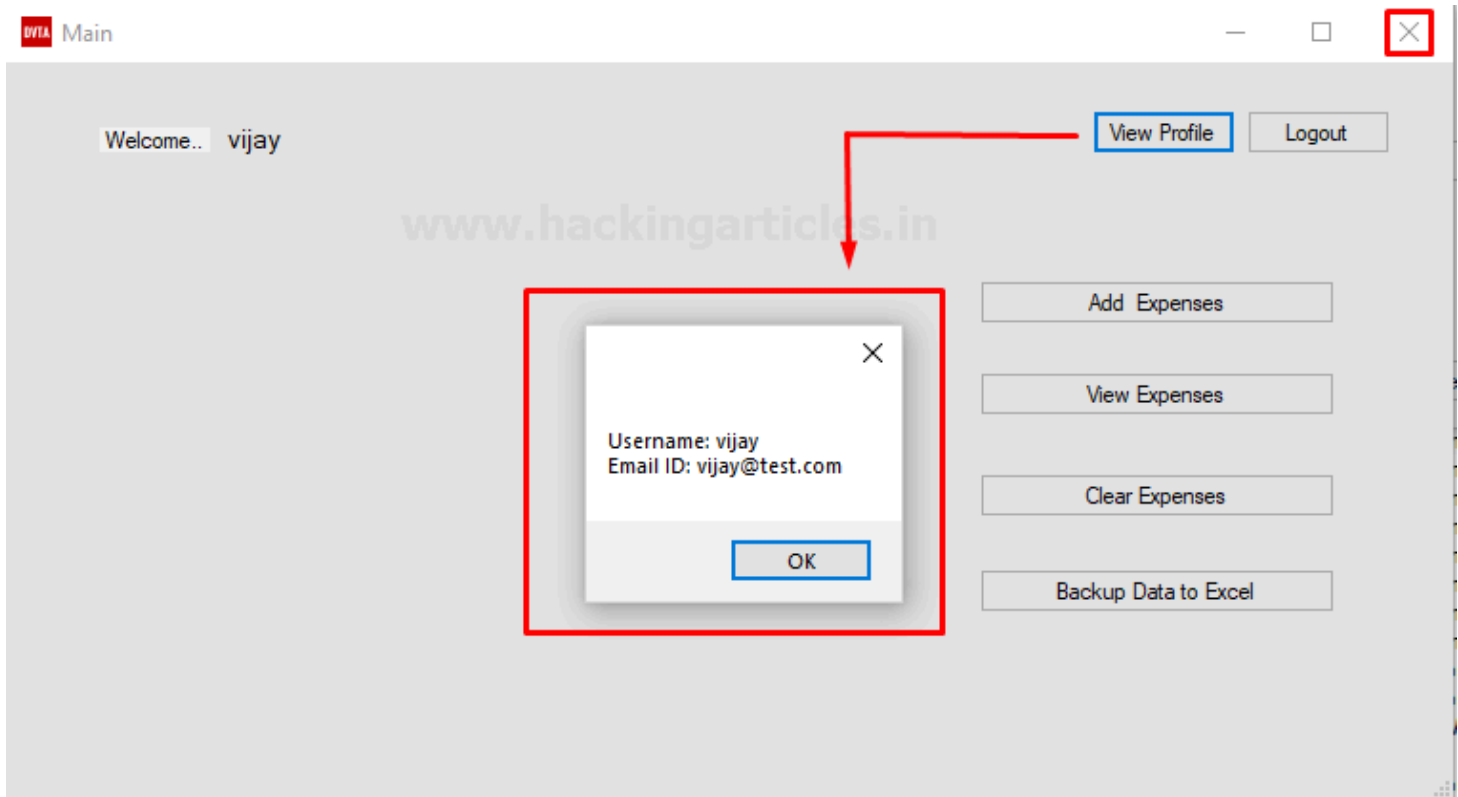
Configure Server

Process Monitor - Sysinternals: www.sysinternals.com

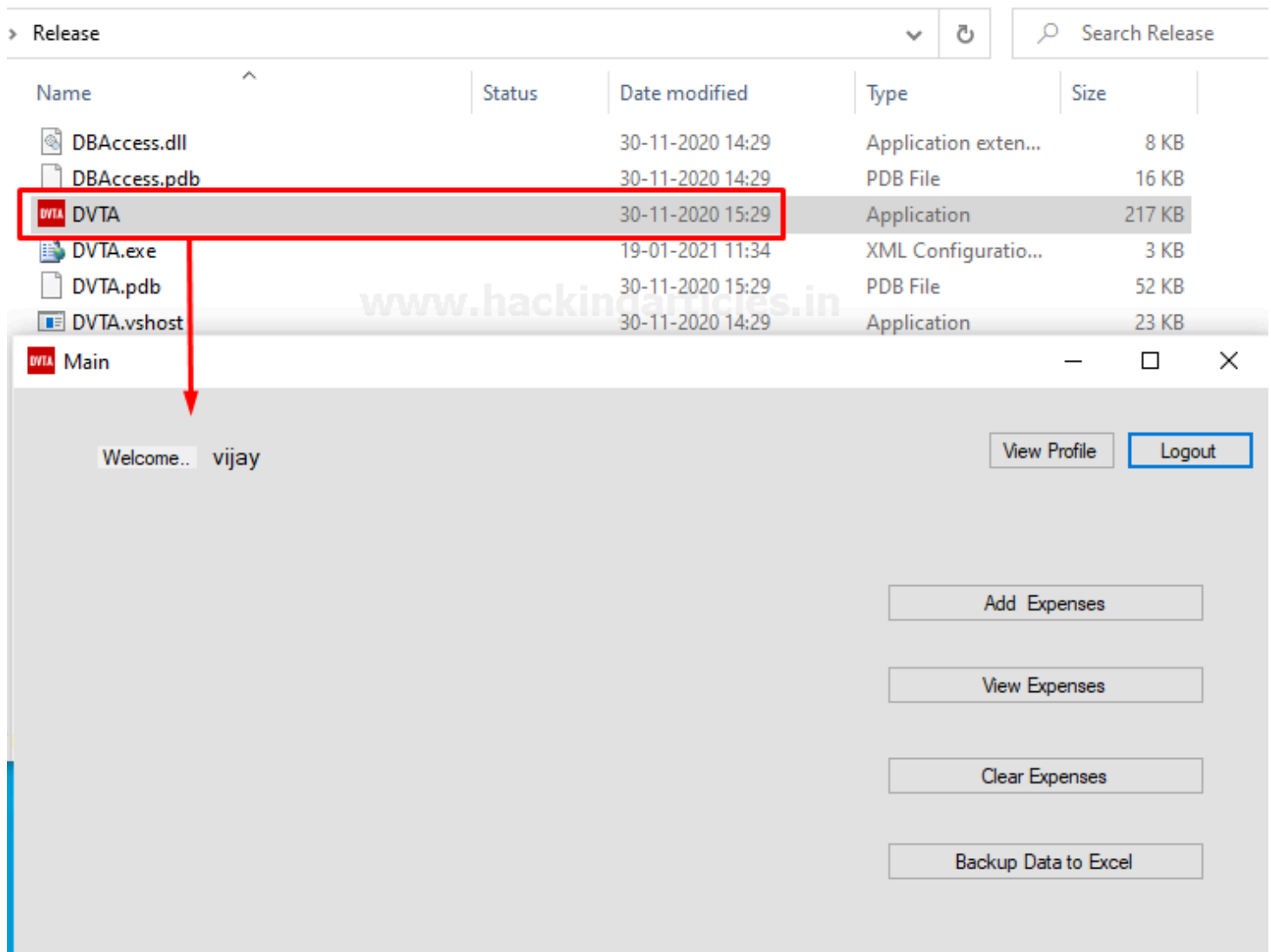
File Edit Event Filter Tools Options Help

Time ...	Process Name	PID	Operation	Path
3:30:...	DVTA.exe	6004	RegOpenKey	HKU
3:30:...	DVTA.exe	6004	RegQueryKey	HKU
3:30:...	DVTA.exe	6004	RegQueryKey	HKU
3:30:...	DVTA.exe	6004	RegOpenKey	HKU\DEFAULT
3:30:...	DVTA.exe	6004	RegSetInfoKey	HKU\DEFAULT
3:30:...	DVTA.exe	6004	RegQueryKey	HKU\DEFAULT
3:30:...	DVTA.exe	6004	RegOpenKey	HKU\DEFAULT\Software\Microsoft
3:30:...	DVTA.exe	6004	RegCloseKey	HKU\DEFAULT
3:30:...	DVTA.exe	6004	RegQueryValue	HKU\DEFAULT\Software\Microsoft
3:30:...	DVTA.exe	6004	RegQueryKey	HKLM
3:30:...	DVTA.exe	6004	RegQueryKey	HKLM
13:30:...	DVTA.exe	6004	RegOpenKey	HKLM\Software\Wow6432N...

As you can see, we have successfully logged in to the application also you can verify the user credentials by navigating to View profile. Now close the tab directly without doing Logout.

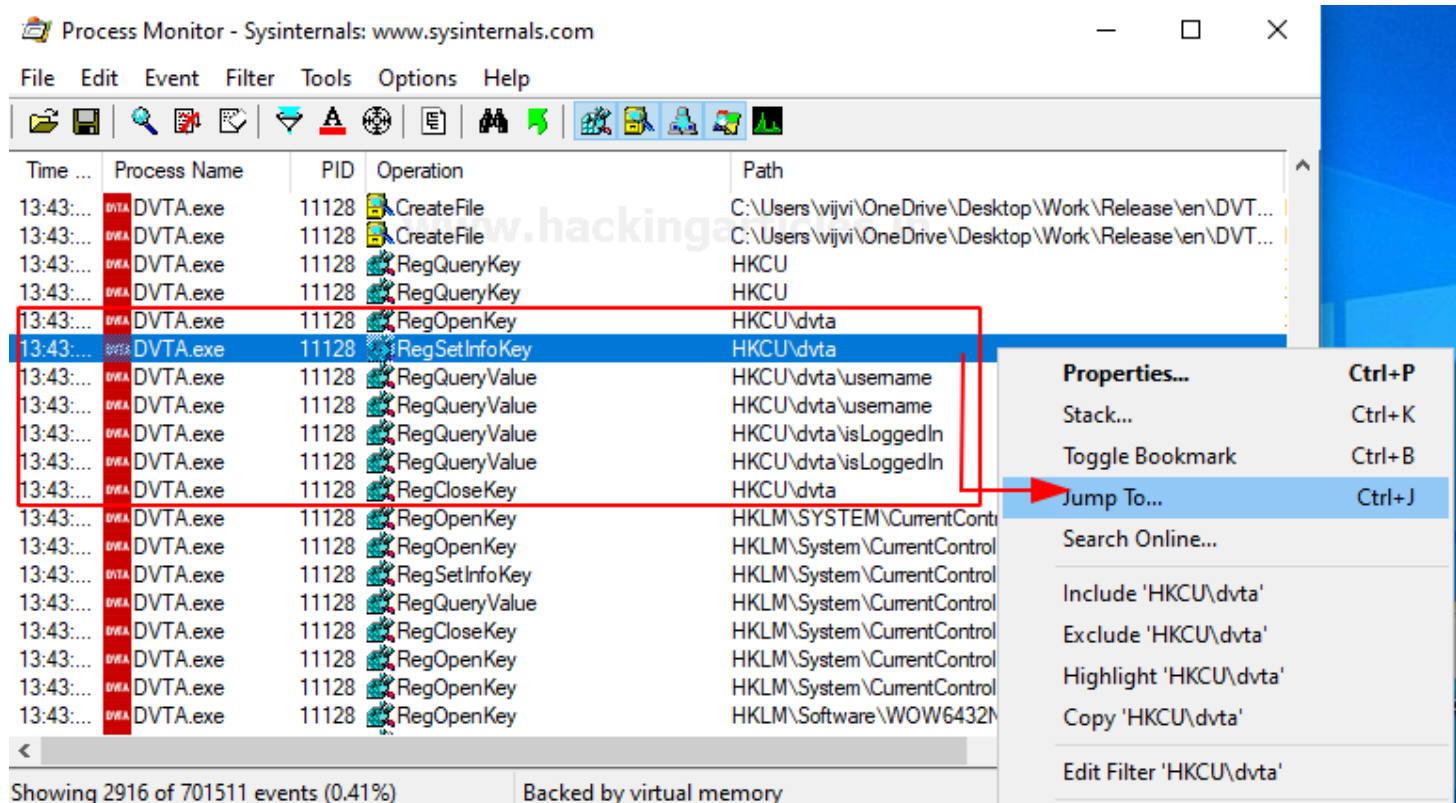


Now again open the DVTA application and if you noticed it doesn't ask to enter the credentials again. That means it is storing the credentials somewhere or it may be storing the session somewhere in the client. So, what we will try to do is we will close the application and clear the procmon display once again so that we'll have to deal with less number of entries and once again we'll open DVTA.exe and we already notice we didn't Logout so it will directly bring you to the main screen

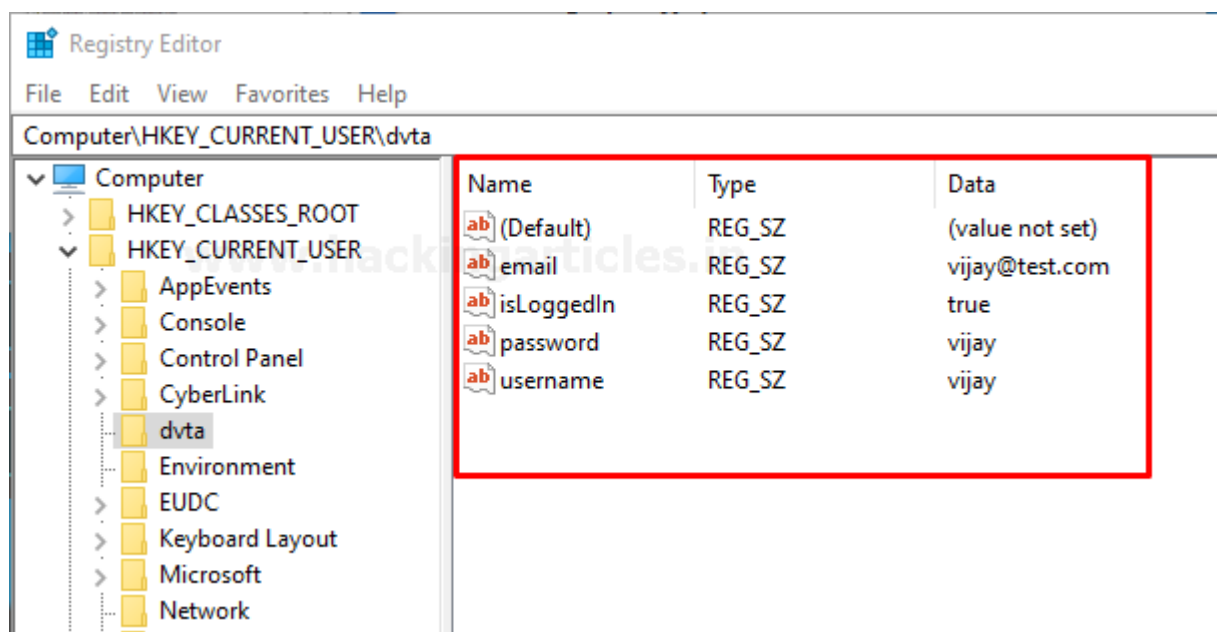


Now, let's see if there is anything interesting happening on the file system when we trying to open this application with the existing session. So, keep scrolling down to the output screen until we don't find something interesting and if you see there is a **registry specific to our application name DVTA and inside that there seem to be some keys like username, is Logged in... etc**

So, it seems like interesting let's quickly open up the registry entry by giving it a **right-click and navigating to Jump to** or you can directly **jump to** file by pressing **ctrl+J**.

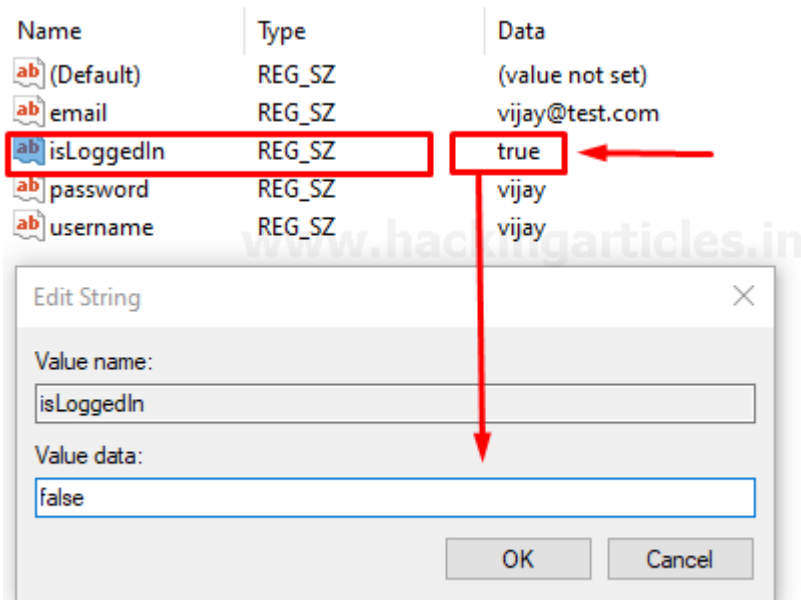


It will open up the Registry entry. As we can see it has a couple of keys like **Email, Is Logged in, Password, and Username**. So basically, this application is storing the credentials here and it is also checking if the user is already logged in or not.



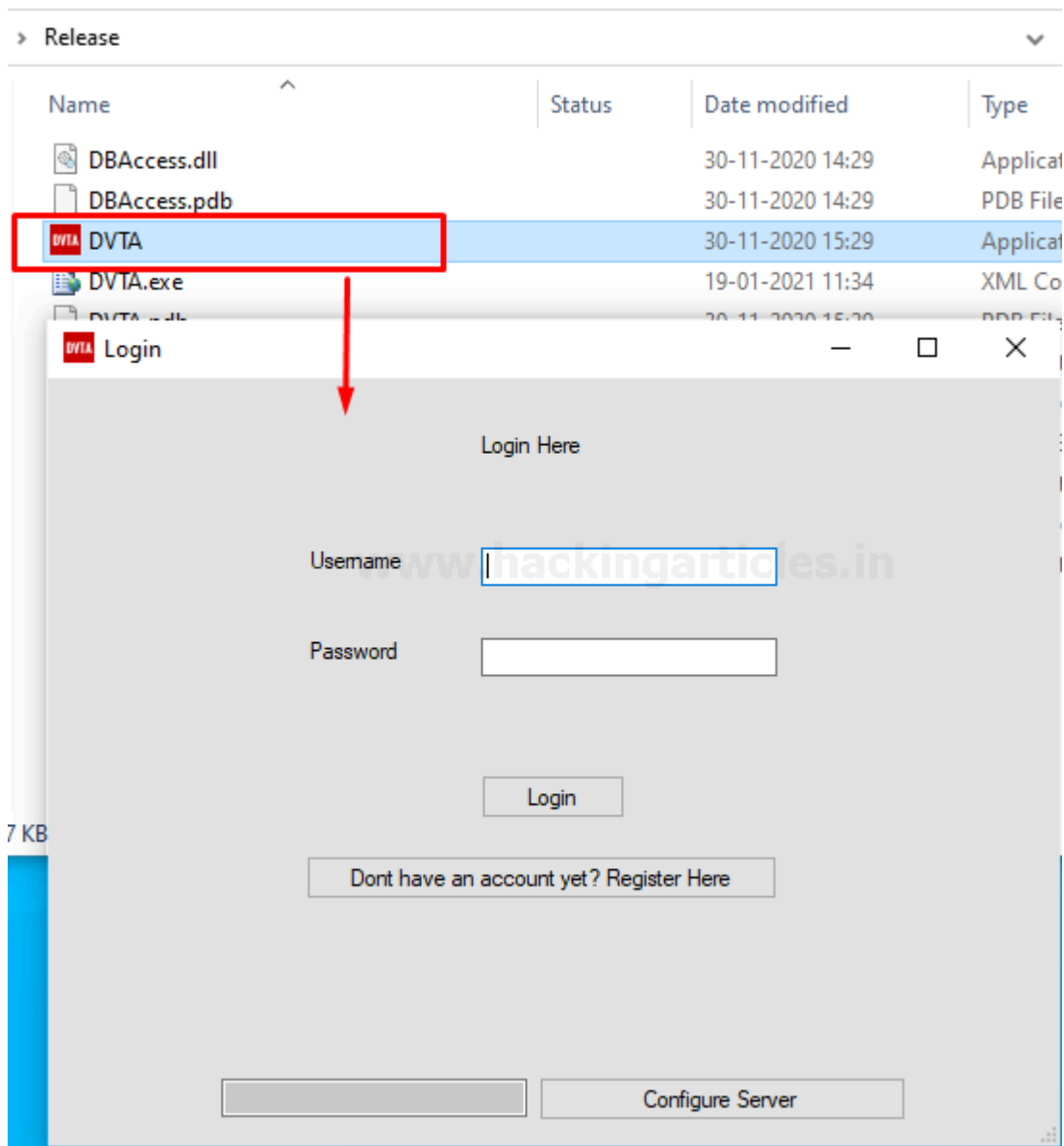
And that means if you change the isLoggedIn value true to false then it will probably ask you to enter the login credentials again even if you are already logged in.

Hmm... exited Let's do it.



Let's check that by closing the DVTA application and opening it once again.

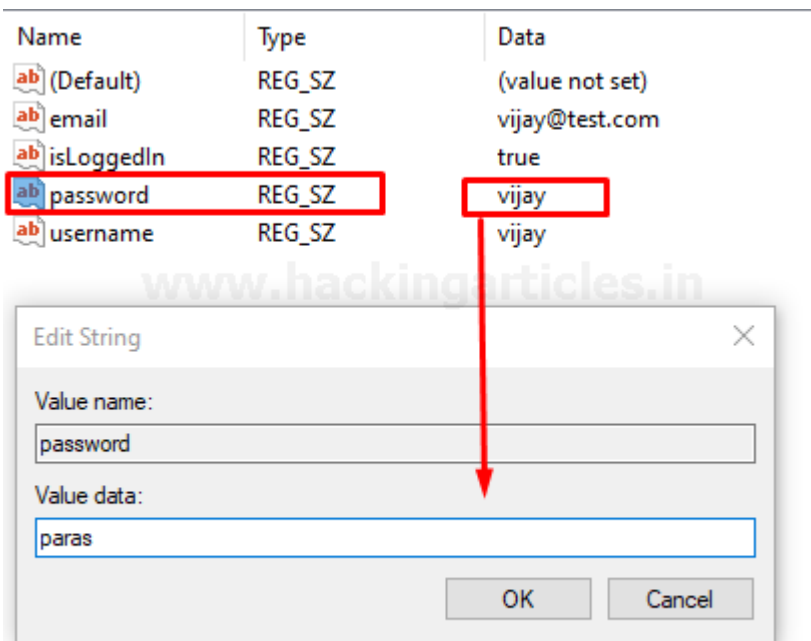
When you open the DVTA application again it will ask you to enter the login credentials again so that means our acknowledges are correct. If the application finds this registry entry with the value true it will not ask you for login.



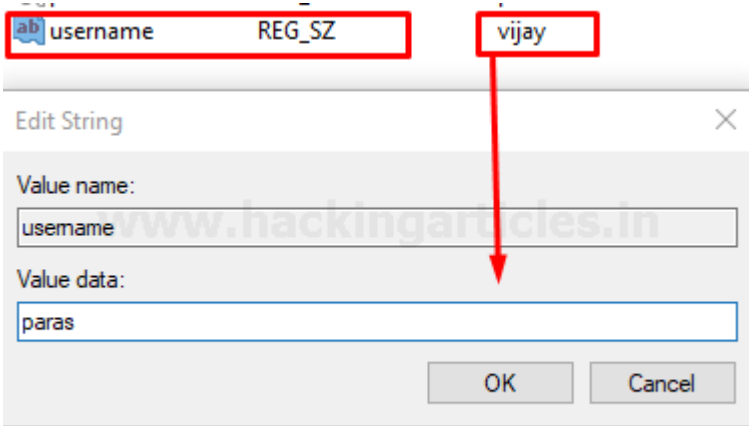
So, this can be used for some interesting attacks for example if we change the user name and password In the registry entry, we can easily trick to the application to believe that the particular user is already logged in with his credentials.

This looks very interesting attack ... isn't it...?? Let's do it

Change the Password value to Vijay to paras or you're on own as by giving a double click to the registry of Password as shown below.

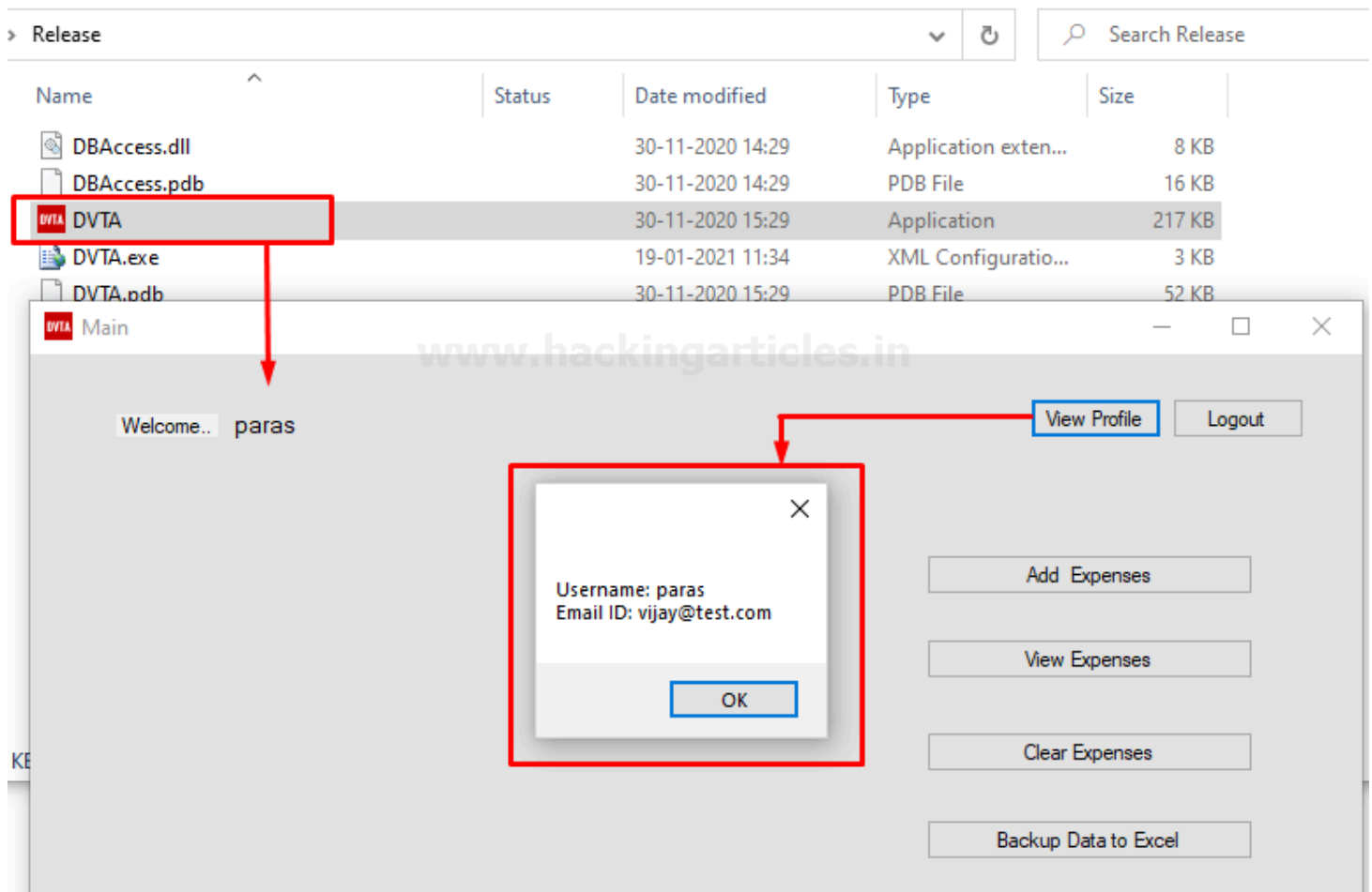


Same again change the username value vijay to paras or your own name and hit ok



Let's check what happens if we once again open the DVTA application.

As you can see, we directly logged in to the application without entering the login credentials. So, as we see we can simply trick the application to logged in to application directly by doing changes in the registry entry.



Now what we learnt from all of these processes that the application is storing login credentials in the registry entry.

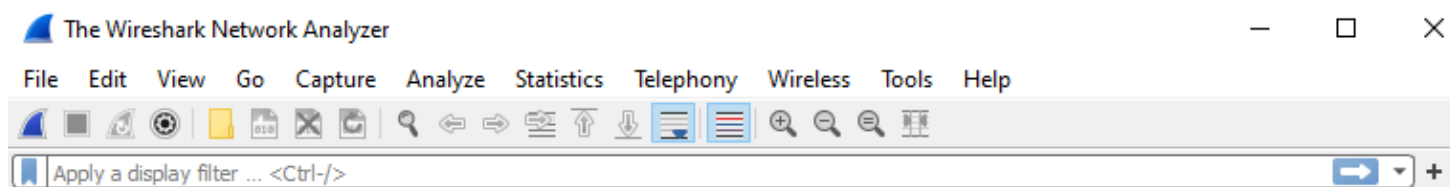
Information gathering by using Wireshark

You can download Wireshark by visiting the official page of Wireshark: –

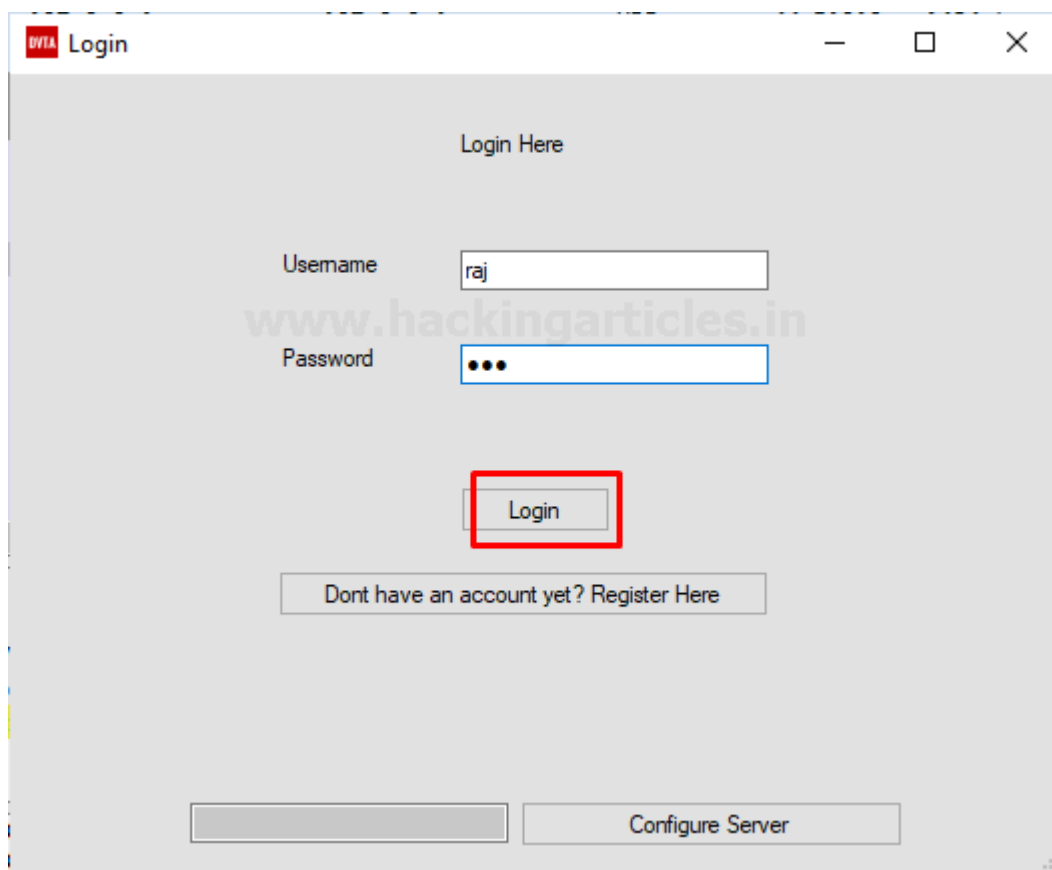
<https://www.wireshark.org/download.html>

After downloading the Wireshark install it on your system by hitting next and next it doesn't need any changes in the installation. By installing it run the Wireshark and the interface looks similar like as shown in below image.

Since we are using loopback address let's choose loopback traffic capture



Now go back to the modified DVTA application and login to the application by using login credentials.



Just after login to application come back to the Wireshark and you can see there are lots of traffic going there just refresh the capture and again login to the DVTA application and just after hitting the login button come back to Wireshark and its better to stop the capture immediately after clicking the login button so it won't capture unnecessary traffic as shown below in image where I opened two different tabs one of Wireshark and another one is DVTA application just to see how it capture the traffic when I press the login button.

The image shows two windows. The top window is Wireshark, titled '*Adapter for loopback traffic capture'. It displays a list of network packets. A red box highlights packets 5 through 12. Below the Wireshark window is the DVTA application window, titled 'DVTA Main'. It shows a login dialog box with the text 'Username: raj' and 'Email ID: raj@test.com' and an 'OK' button. A red arrow points from the 'OK' button in the DVTA login dialog to the highlighted packet 5 in the Wireshark packet list.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	127.0.0.1	127.0.0.1	TCP	49	64690 → 14147 [PSH, ACK] Seq=
2	0.000143	127.0.0.1	127.0.0.1	TCP	44	14147 → 64690 [ACK] Seq=1 Ack
3	0.000362	127.0.0.1	127.0.0.1	TCP	49	14147 → 64690 [PSH, ACK] Seq=
4	0.000439	127.0.0.1	127.0.0.1	TCP	44	64690 → 14147 [ACK] Seq=6 Ack
5	8.005221	127.0.0.1	127.0.0.1	UDP	44	51219 → 1434 Len=12
6	8.080528	127.0.0.1	127.0.0.1	UDP	133	1434 → 51219 Len=101
7	8.081931	127.0.0.1	127.0.0.1	TCP	56	64730 → 58636 [SYN] Seq=0 Win
8	8.082225	127.0.0.1	127.0.0.1	TCP	56	58636 → 64730 [SYN, ACK] Seq=
9	8.082359	127.0.0.1	127.0.0.1	TCP	44	64730 → 58636 [ACK] Seq=1 Ack
10	8.520819	127.0.0.1	127.0.0.1	TLSv1	148	Ignored Unknown Record
11	8.520933	127.0.0.1	127.0.0.1	TCP	44	58636 → 64730 [ACK] Seq=1 Ack
12	8.521056	127.0.0.1	127.0.0.1	TLSv1	87	Ignored Unknown Record

DVTA Main

Welcome.. raj [View Profile](#) [Logout](#)

[Add Expenses](#)

[View Expenses](#)

[Clear Expenses](#)

[Backup Data to Excel](#)

Username: raj
Email ID: raj@test.com

[OK](#)

Now what we're interested in is to understand how this login is making communication with the server. If you look there are some packets that uses UDP protocol so let's open it in UDP stream and then let's see what we can find.

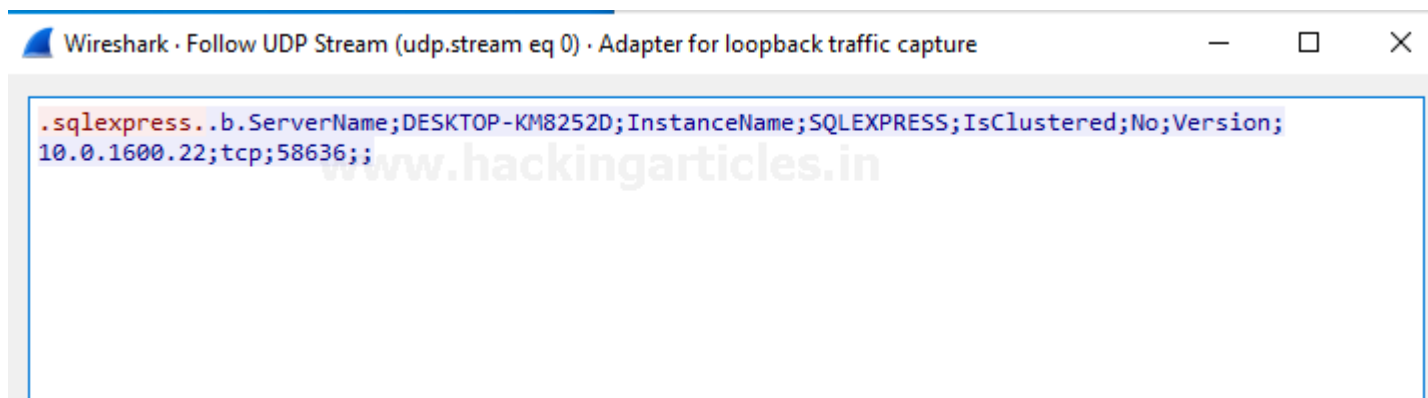
No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	127.0.0.1	127.0.0.1	TCP	49	64690 → 14147 [PSH, ACK] Seq=
2	0.000143	127.0.0.1	127.0.0.1	TCP	44	14147 → 64690 [ACK] Seq=1 Ack
3	0.000362	127.0.0.1	127.0.0.1	TCP	49	14147 → 64690 [PSH, ACK] Seq=
4	0.000439	127.0.0.1	127.0.0.1	TCP	44	64690 → 14147 [ACK] Seq=6 Ack
5	8.005221	127.0.0.1	127.0.0.1	UDP	44	51219 → 1434 Len=12
6	8.080528			UDP	133	1434 → 51219 Len=101
7	8.081931			TCP	56	64730 → 58636 [SYN] Seq=0 Win
8	8.082225			TCP	56	58636 → 64730 [SYN, ACK] Seq=
9	8.082359			TCP	44	64730 → 58636 [ACK] Seq=1 Ack
10	8.520819			TLSv1	148	Ignored Unknown Record
11	8.520933			TCP	44	58636 → 64730 [ACK] Seq=1 Ack
12	8.521056			TLSv1	87	Ignored Unknown Record
13	8.521111			TCP	44	64730 → 58636 [ACK] Seq=105 A
14	8.581366			TLSv1	205	Ignored Unknown Record
15	8.581466			TCP	44	58636 → 64730 [ACK] Seq=44 Ac
16	8.595212			TLSv1	902	Ignored Unknown Record
17	8.595318			TCP	44	64730 → 58636 [ACK] Seq=266 A
18	8.622368			TLSv1	218	Ignored Unknown Record
19	8.622409			TCP	44	58636 → 64730 [ACK] Seq=902 A

Mark/Unmark Packet	Ctrl+M
Ignore/Unignore Packet	Ctrl+D
Set/Unset Time Reference	Ctrl+T
Time Shift...	Ctrl+Shift+T
Packet Comment...	Ctrl+Alt+C
Edit Resolved Name	
Apply as Filter	▶
Prepare as Filter	▶
Conversation Filter	▶
Colorize Conversation	▶
SCTP	▶

Follow	TCP Stream	Ctrl+Alt+Shift+T	IPF_Loopback, id 0
	UDP Stream	Ctrl+Alt+Shift+U	
	TLS Stream	Ctrl+Alt+Shift+S	

When you open that packet, you can see this is a packet containing a request to SQL Server. You can see there is an SQL Express and the hostname where the SQL server is running.

So, this packet shows that we are making a network connection with the SQL server and this is enough also you can capture the whole traffic of DVTA application how the application is communicating with the server.



Now the key takeaway here is that the application is making some SQL connections when you are trying to interact with it.

In the next part, we will talk about different attack vectors.