

Hackerman's Hacking Tutorials

The knowledge of anything, since all things have causes, is not acquired or complete unless it is known by its causes. - Avicenna

[About Me!](#)[Cheat Sheet](#)[My Clone](#)[How This Website is Built](#)[The Other Guy from Wham!](#)

JUL 15, 2018 - 8 MINUTE READ - [COMMENTS](#) -

[REVERSE ENGINEERING](#)[DVTA](#)[WRITEUP](#)

DVTA - Part 1 - Setup

- [Existing Setup Instructions](#)
- [Setup Instructions 2: Electric Boogaloo](#)
 - [0. Ingredients and Price](#)
 - [1. Get the Code and Binary](#)
 - [2. Install Microsoft SQL Server 2008 Express](#)
 - [3. Install Microsoft SQL Server 2008 Management Studio Express](#)
 - [4. Create the DVTA Database](#)
 - [5. Setup the FTP Server](#)
 - [6. Modify DVTA to Connect to Our Local SQL Server](#)
 - [7. Fix the FTP Connectivity](#)
 - [7.1 Use dnSpy to Modify the Hardcoded FTP Address](#)

Who am I?

I am Parsia, a security engineer at [Electronic Arts](#).

I write about application security, reverse engineering, Go, cryptography, and (obviously) videogames.

Click on [About Me!](#) to know more.



in

Collections

- [Discover the FTP Address](#)
- [Modify the Address in Binary](#)
- [Conclusion](#)

I have written a lot about thick clients. However, I have not done more than a few practical examples that I can show my co-workers or anyone else asking questions. Recently, I came across the Damn Vulnerable Thick Client Application by SecVulture at <https://github.com/secvulture/dvta>.

I am not going to use the original version of the application. Someone has created a fork and added more protections. We will use this fork instead:

- <https://github.com/nddmars/dvta>

Neither fork's setup instructions worked for me. As a result, the first part is actually setting up the application and the necessary back-end in only one VM. But don't worry, we will do a bit of reverse engineering with dnSpy to fix an issue.

Thanks to SecVulture for creating the app and maintainers of the second repository for adding protections.

Existing Setup Instructions

There are no instructions in the original repository at:

- <https://github.com/secvulture/dvta>

But author's has some post on Infosec Institute with setup and solutions at [1](#):

[Thick Client Proxying](#)

[Go/Golang](#)

[Blockchain/Distributed Ledgers](#)

[Automation](#)

[Reverse Engineering](#)

[Crypto\(graphy\)](#)

[CTFs/Writeups](#)

[WinAppDbg](#)

[AWSome.pw - S3 bucket squatting - my very legit branded vulnerability](#)

- <https://resources.infosecinstitute.com/practical-thick-client-application-penetration-testing-using-damn-vulnerable-thick-client-app-part-1>

[The fork](#) has a Word document file with pictures and setup instructions. I still could not make it work.

Setup Instructions 2: Electric Boogaloo

I know setup is boring and you want to "hack." But this is necessary to have fun later.

0. Ingredients and Price

Hint: Everything is free.

1. Windows 7 (or 10) VM. I used a 32-bit Windows 7 VM from <https://modern.ie>: Free.
2. Microsoft SQL Server 2008 Express: Free.
3. Microsoft SQL Server 2008 Management Studio Express: Free.
4. FileZilla FTP Server: Free.
5. Microsoft Sysinternals Suite: Free.
6. dnSpy: Free.

1. Get the Code and Binary

Download the whole repository as a zip file (because you don't want to install git on a disposable VM like me) from:

- <https://github.com/nddmars/dvta>

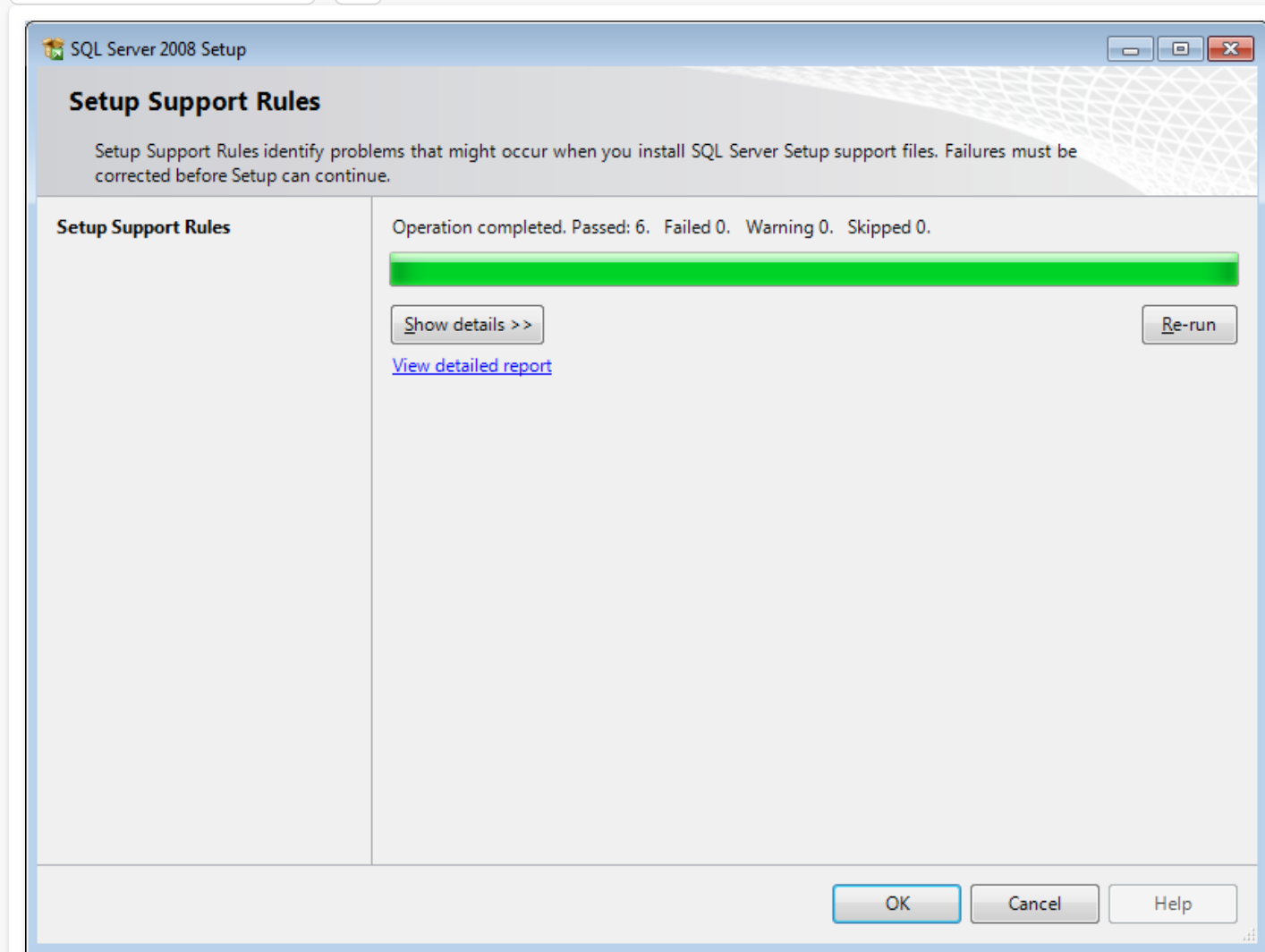
Extract it to a location of your choice. I named mine `dvta-master`.

2. Install Microsoft SQL Server 2008 Express

- Download it from <https://www.microsoft.com/en-us/download/confirmation.aspx?id=1695>.
- Click on `Installation` to the left and select `New SQL Server stand-alone ...`.

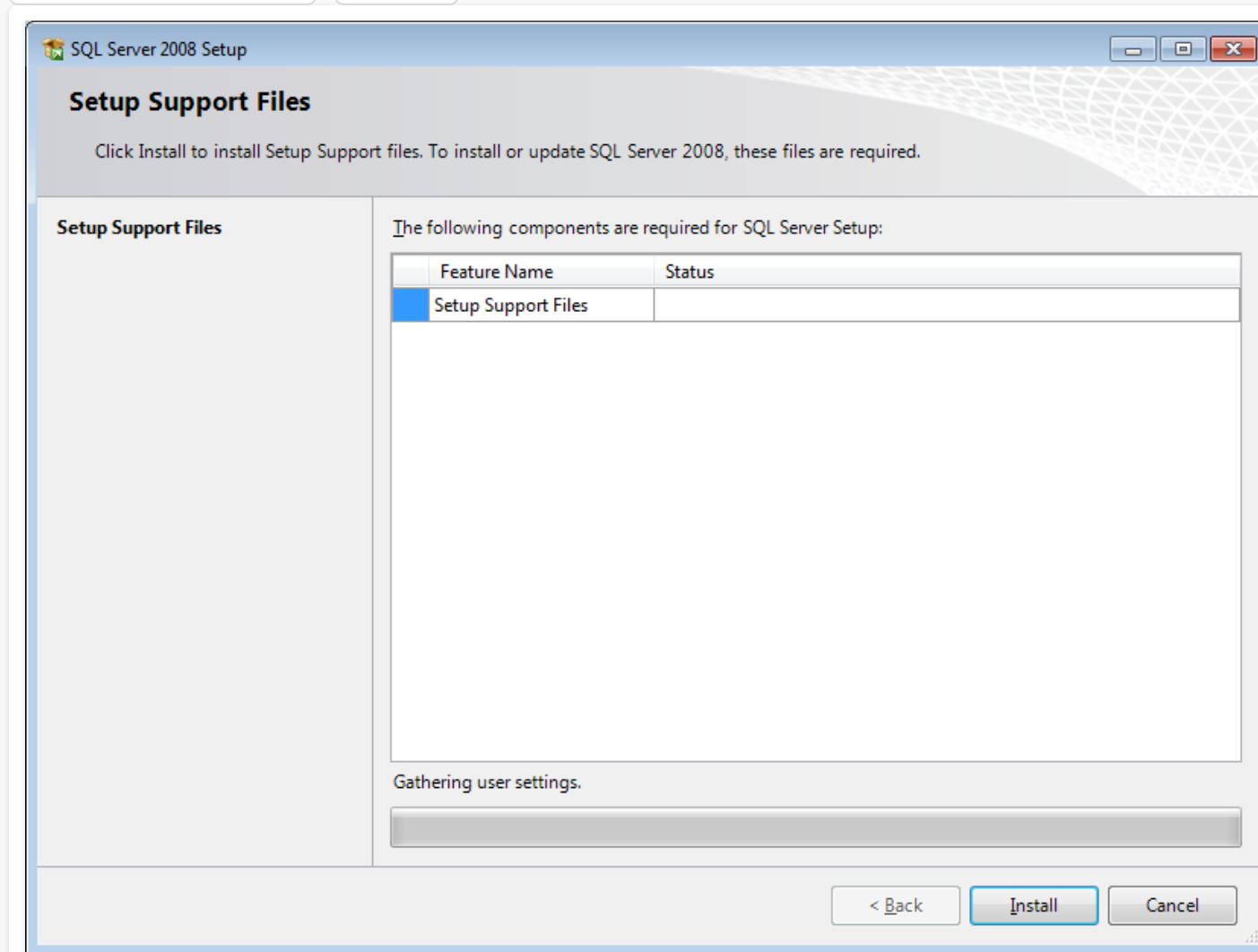


- Setup Support Rules : OK.



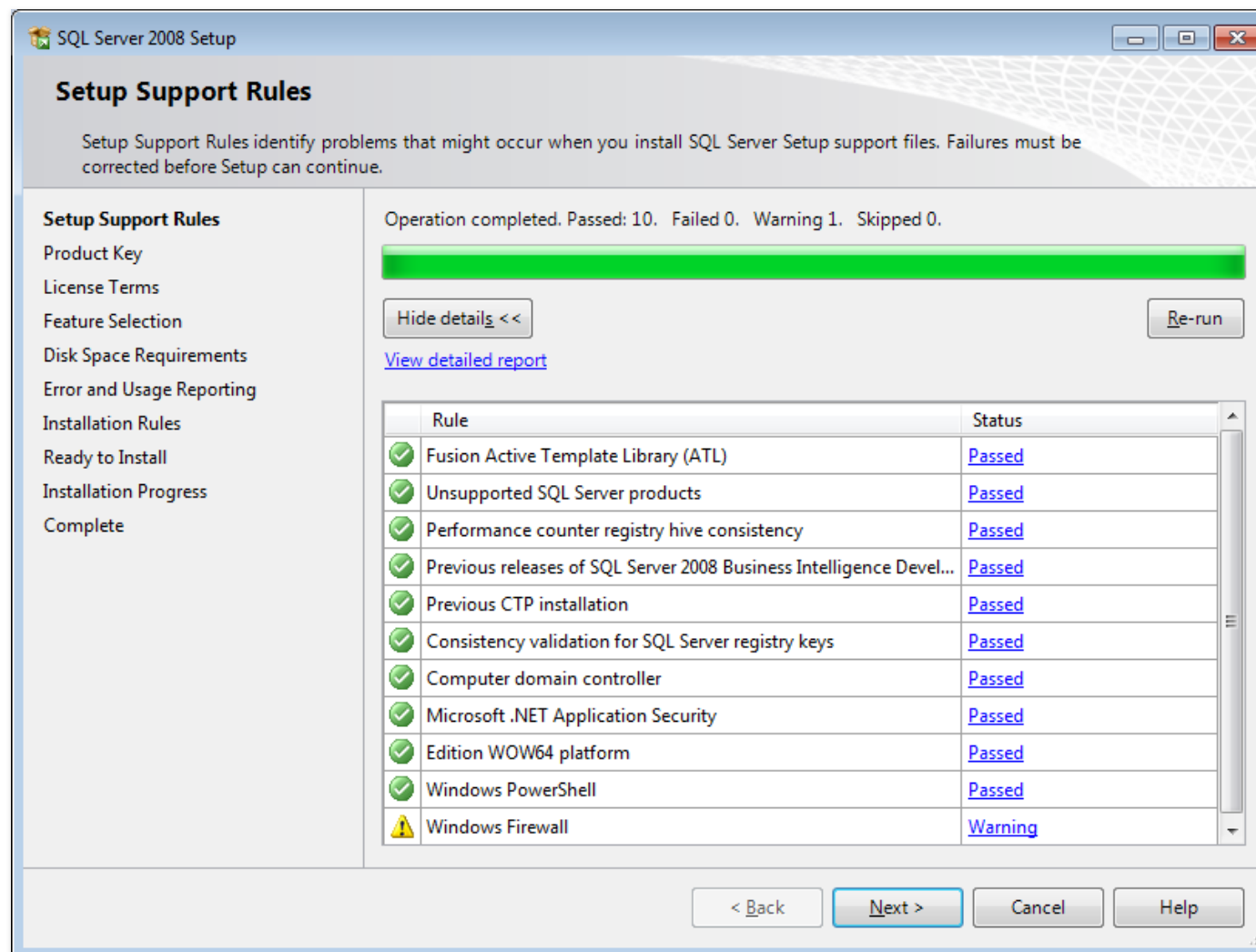
Support Rules will Run

- Setup Support Files: Install.



Select install to get setup support files

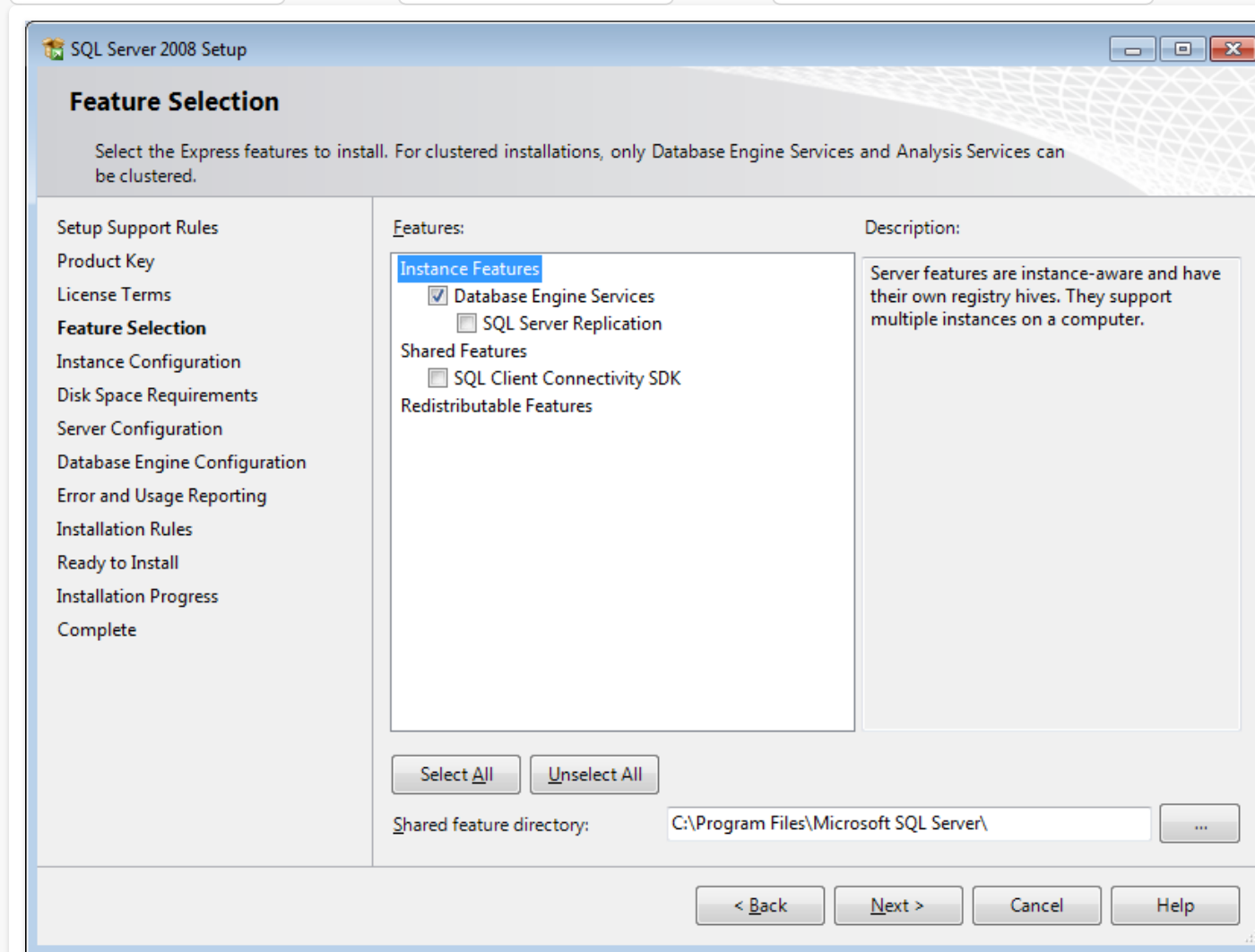
- Again in **Setup Support Files** : **Next** .



Ignore the Firewall warning, our back-end is local

- **Product Key** : Continue with free edition.
- **License Terms** : **Accept** .

- Feature Selection : Under Instance Features select Database Engine Services .



We do not need the SDK

- **Instance Configuration**: Keep the default instance name **SQLExpress**.

The screenshot shows the 'Instance Configuration' window of the SQL Server 2008 Setup. The window title is 'SQL Server 2008 Setup'. The main heading is 'Instance Configuration' with the instruction 'Specify the name and instance ID for the SQL Server instance.' On the left is a navigation pane with the following items: Setup Support Rules, Product Key, License Terms, Feature Selection, **Instance Configuration**, Disk Space Requirements, Server Configuration, Database Engine Configuration, Error and Usage Reporting, Installation Rules, Ready to Install, Installation Progress, and Complete. The main area has two radio buttons: 'Default instance' (unselected) and 'Named instance:' (selected). The 'Named instance:' text is followed by a text box containing 'SQLExpress'. Below this, there are three text boxes: 'Instance ID:' containing 'SQLExpress', 'Instance root directory:' containing 'C:\Program Files\Microsoft SQL Server\' with a browse button (...), and 'SQL Server directory:' containing 'C:\Program Files\Microsoft SQL Server\MSSQL10.SQLExpress'. At the bottom, it says 'Installed instances:' followed by a table with the following columns: Instance, Features, Edition, Version, and Instance ID. The table is currently empty. At the bottom right of the window are four buttons: '< Back', 'Next >', 'Cancel', and 'Help'.

If you change the default instance name, replace it in the rest of the instructions.

- **Disk Space Requirements** **Next**.

- Server Configuration: I selected the SYSTEM account for SQL Server Database Engine. Change SQL Server Browser to Automatic.

SQL Server 2008 Setup

Server Configuration

Specify the configuration.

- Setup Support Rules
- Product Key
- License Terms
- Feature Selection
- Instance Configuration
- Disk Space Requirements
- Server Configuration**
- Database Engine Configuration
- Error and Usage Reporting
- Installation Rules
- Ready to Install
- 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 AUTHORITY\SYSTEM		Automatic ▼
SQL Server Browser	NT AUTHORITY\LOCAL SERVICE		Automatic ▼

Use the same account for all SQL Server services

These services will be configured automatically where possible to use a low privilege account. On some older Windows versions the user will need to specify a low privilege account. For more information, click Help.

< Back Next > Cancel Help

Doesn't really matter if we use a privileged account in a VM

- Database Engine Configuration : Under Authentication Mode select Mixed Mode ... and enter p@ssw0rd as password. Then Add Current User .

SQL Server 2008 Setup

Database Engine Configuration

Specify Database Engine authentication security mode, administrators and data directories.

Setup Support Rules
Product Key
License Terms
Feature Selection
Instance Configuration
Disk Space Requirements
Server Configuration
Database Engine Configuration
Error and Usage Reporting
Installation Rules
Ready to Install
Installation Progress
Complete

Account Provisioning | 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)

Built-in SQL Server system administrator account

Enter password:

Confirm password:

Specify SQL Server administrators

IE11WIN7\IEUser (IEUser)

SQL Server administrators have unrestricted access to the Database Engine.

Add Current User | Add... | Remove

< Back | Next > | Cancel | Help

It appears adding another user is mandatory during setup

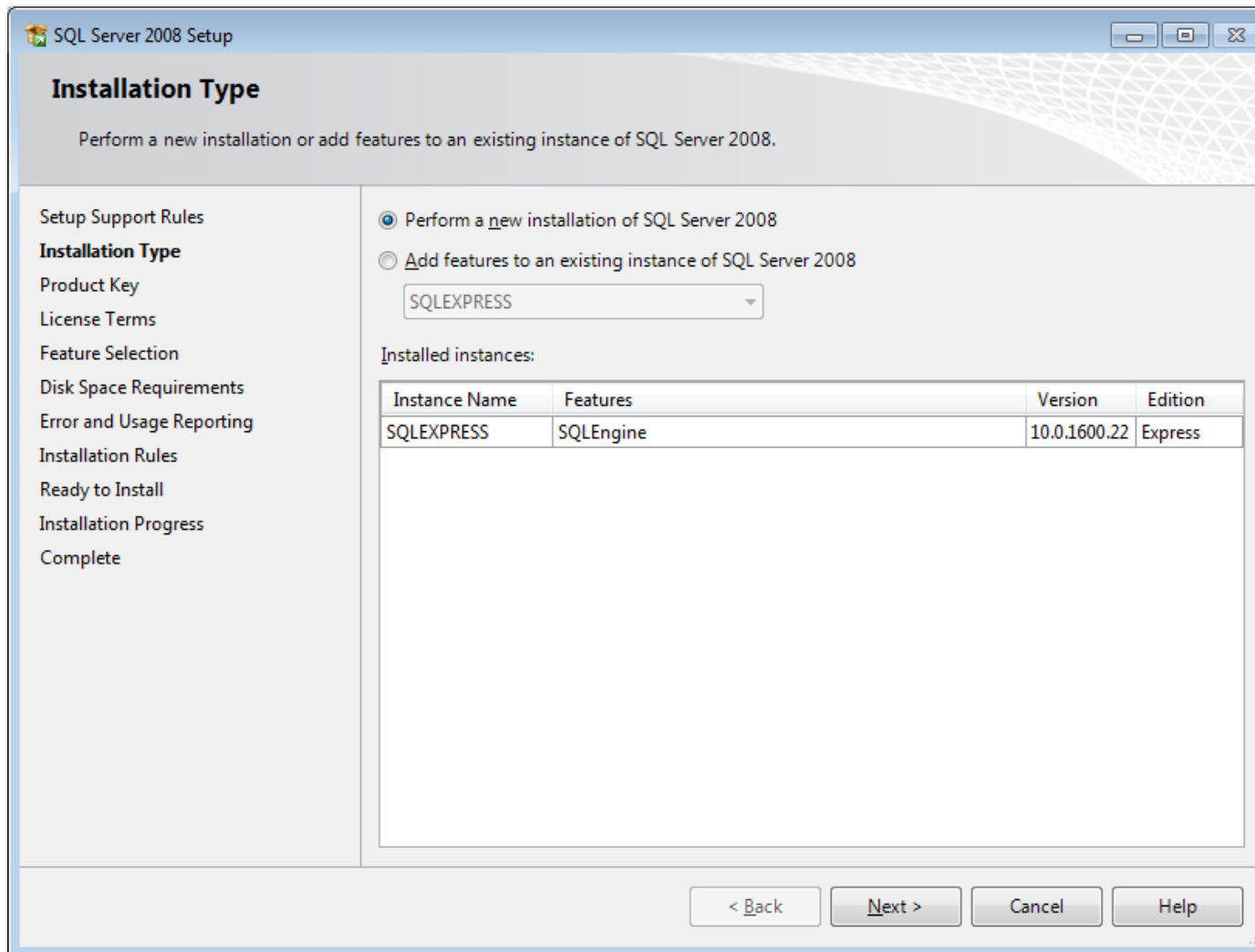
- Error and Usage Reporting : Keep boxes unchecked or don't.

- : .
- : .
- Finally .

3. Install Microsoft SQL Server 2008 Management Studio Express

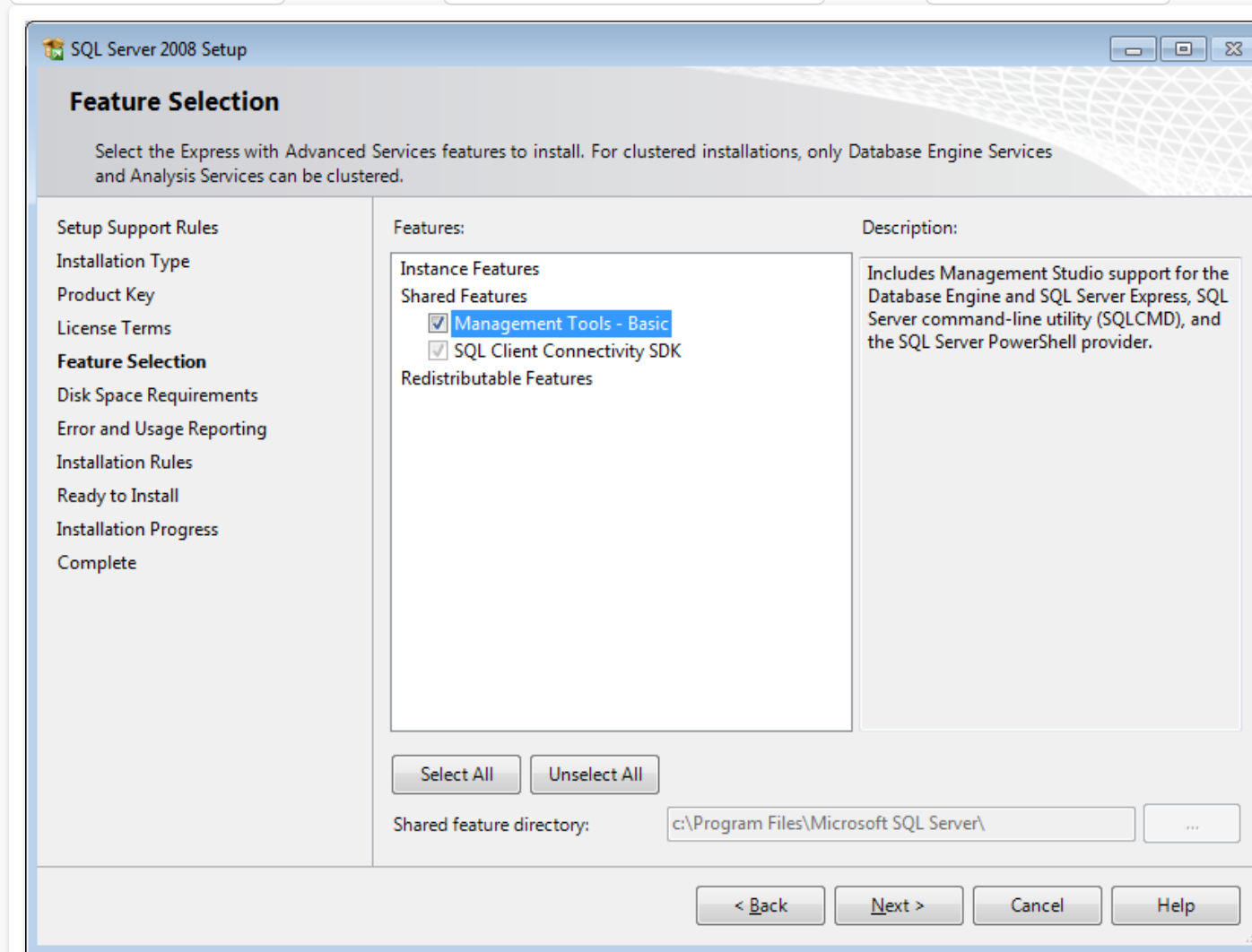
We need management studio to set up our database and tables.

- Download from: <https://www.microsoft.com/en-us/download/details.aspx?id=7593>.
- Ignore the error about Service Pack.
- Click on to the left and select (this looks very similar to last wizard).
- : Select , otherwise the management tools will not show up.



Don't worry, it will not install a new instance

- Feature Selection and select Management Tools - Basic under Shared Features.



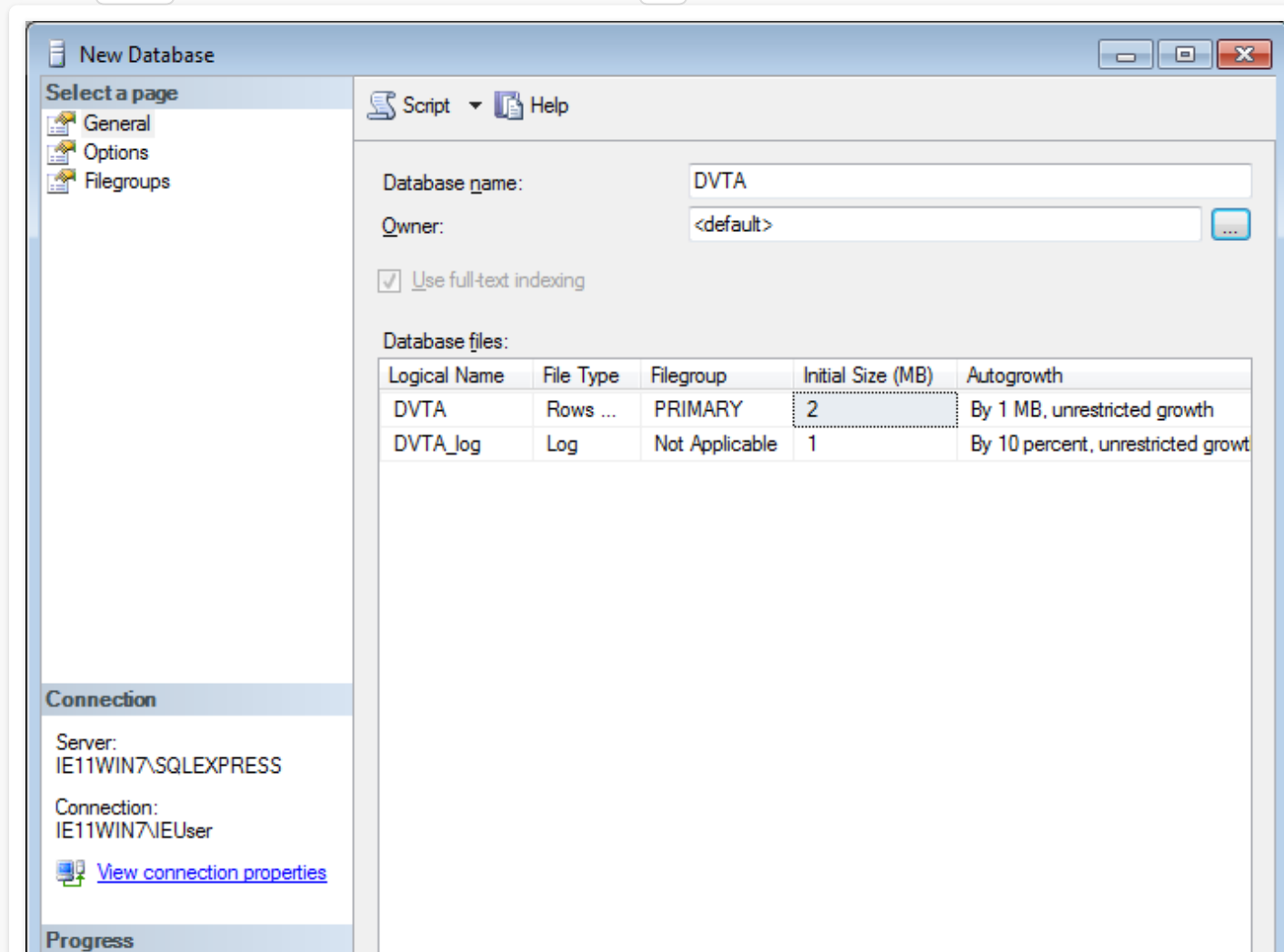
Add Management Studio here

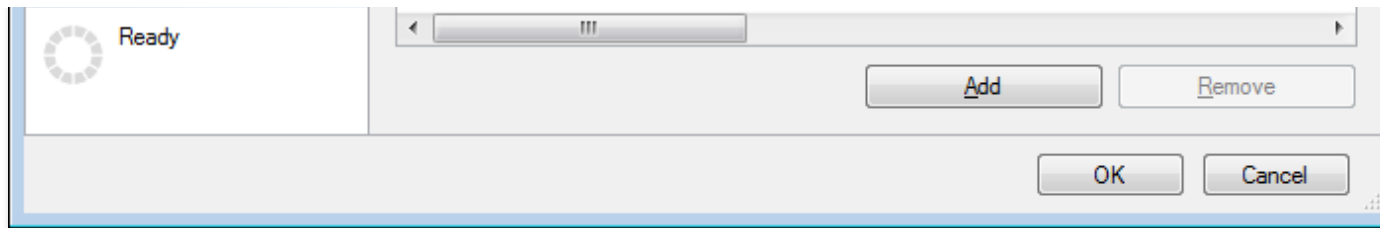
- Complete the installation.

4. Create the DVTA Database

Now we can use the management studio to create the database and populate it.

- Start `SQL Server Management Studio` and connect to the `SQLExpress` instance.
- Right-click on `Databases` to the left and select `New Database`.
- Enter `DVTA` in the database name and press `OK`. Don't change anything else.



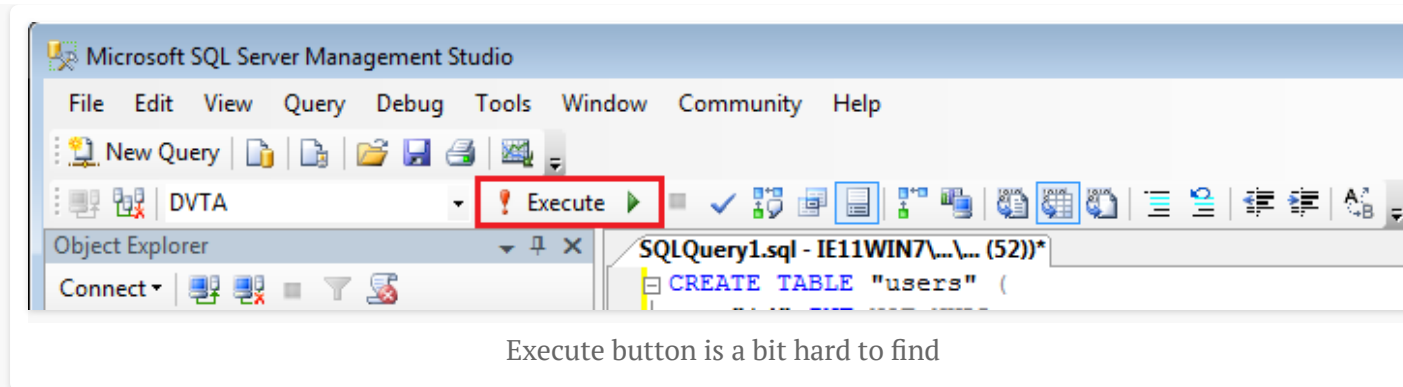


Only change the database name

- Right-click on **DVTAs** under **Databases** and select **New Query**.
- To create the **users** table, enter this query and select **Execute** (note this is different from the original instructions, we are setting the **id** column to auto-increment by **1** starting from **0**). Without auto-increment, registration will not work:

Creating the users table

```
1 CREATE TABLE "users" (  
2     "id" INT IDENTITY(0,1) NOT NULL,  
3     "username" VARCHAR(100) NOT NULL,  
4     "password" VARCHAR(100) NOT NULL,  
5     "email" VARCHAR(100) NULL DEFAULT NULL,  
6     "isadmin" INT NULL DEFAULT '0',  
7     PRIMARY KEY ("id")  
8 )
```

Execute button is a bit hard to find

- Next create the `expenses` table (I have set the `id` column to auto-increment):

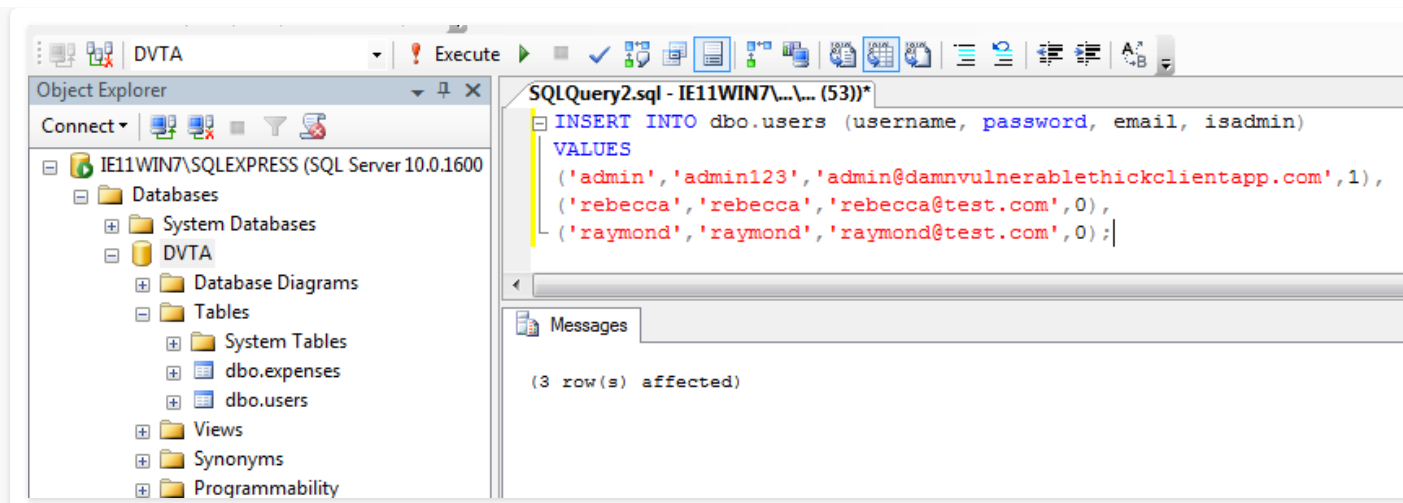
Creating the expenses table

```
1 CREATE TABLE "expenses" (  
2     "id" INT IDENTITY(0,1) NOT NULL,  
3     "email" VARCHAR(100) NOT NULL,  
4     "item" VARCHAR(100) NOT NULL,  
5     "price" VARCHAR(100) NOT NULL,  
6     "date" VARCHAR(100) NOT NULL,  
7     "time" VARCHAR(100) NULL DEFAULT NULL,  
8     PRIMARY KEY ("id")  
9 )
```

- Populate the users table with some test data. The non-admin users can be added through the application later but admin needs to be setup manually.

Adding test users

```
1 INSERT INTO dbo.users (username, password, email, isadmin)  
2 VALUES  
3 ('admin','admin123','admin@damnvulnerablethickclientapp.com',1),  
4 ('rebecca','rebecca','rebecca@test.com',0),  
5 ('raymond','raymond','raymond@test.com',0);
```



Three test users added

- Now we can right click on `dbo.users` and select `Select Top 1000 Rows` to see the test data.

Object Explorer

Connect

IE11WIN7\SQLEXPRESS (SQL Server 10.0.1600 - IE11WIN7)

Databases

System Databases

DVTA

Database Diagrams

Tables

System Tables

dbo.expenses

dbo.users

Views

Synonyms

Programs

Services

Security

Server Objects

Replication

New Table...

Design

Select Top 1000 Rows

Edit Top 200 Rows

Script Table as

View Dependencies

Results

Messages

SQLQuery5.sql - IE11WIN7\...r (52)

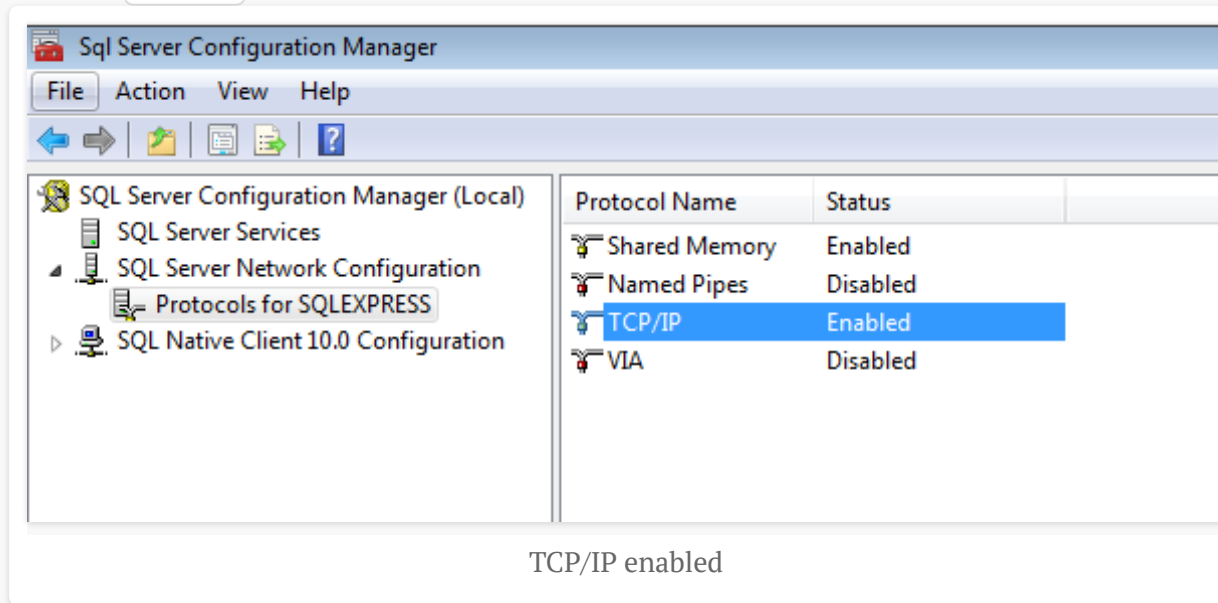
```
/****** Script for SelectTopNRows command from SSMS *****  
SELECT TOP 1000 [id]  
    , [username]  
    , [password]  
    , [email]  
    , [isadmin]  
FROM [DVTA] . [dbo] . [users]
```

	id	username	password	email	isadmin
1	0	admin	admin123	admin@damnvulnerablethickclientapp.com	1
2	1	rebecca	rebecca	rebecca@test.com	0
3	2	raymond	raymond	raymond@test.com	0

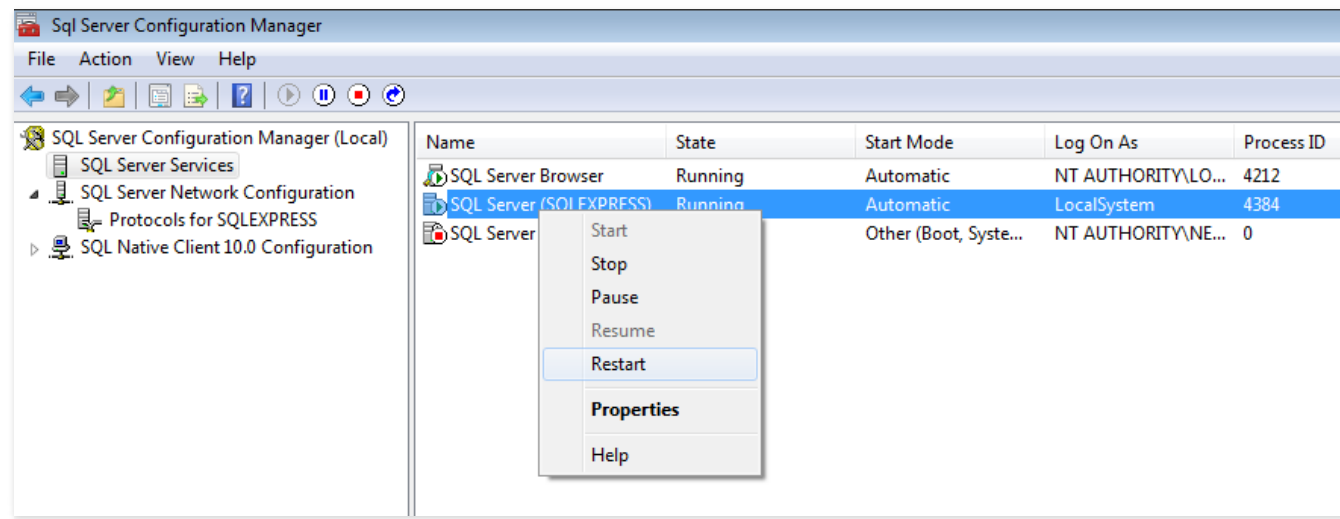
Test users in the database

- Open `SQL Server Configuration Manager` and click on `SQL Server Network Configuration > Protocols for SQLEXPRESS`

- Enable TCP/IP .



- After enabling TCP/IP , you need to restart the SQL Server (SQLEXPRESS) service under SQL Server Services .

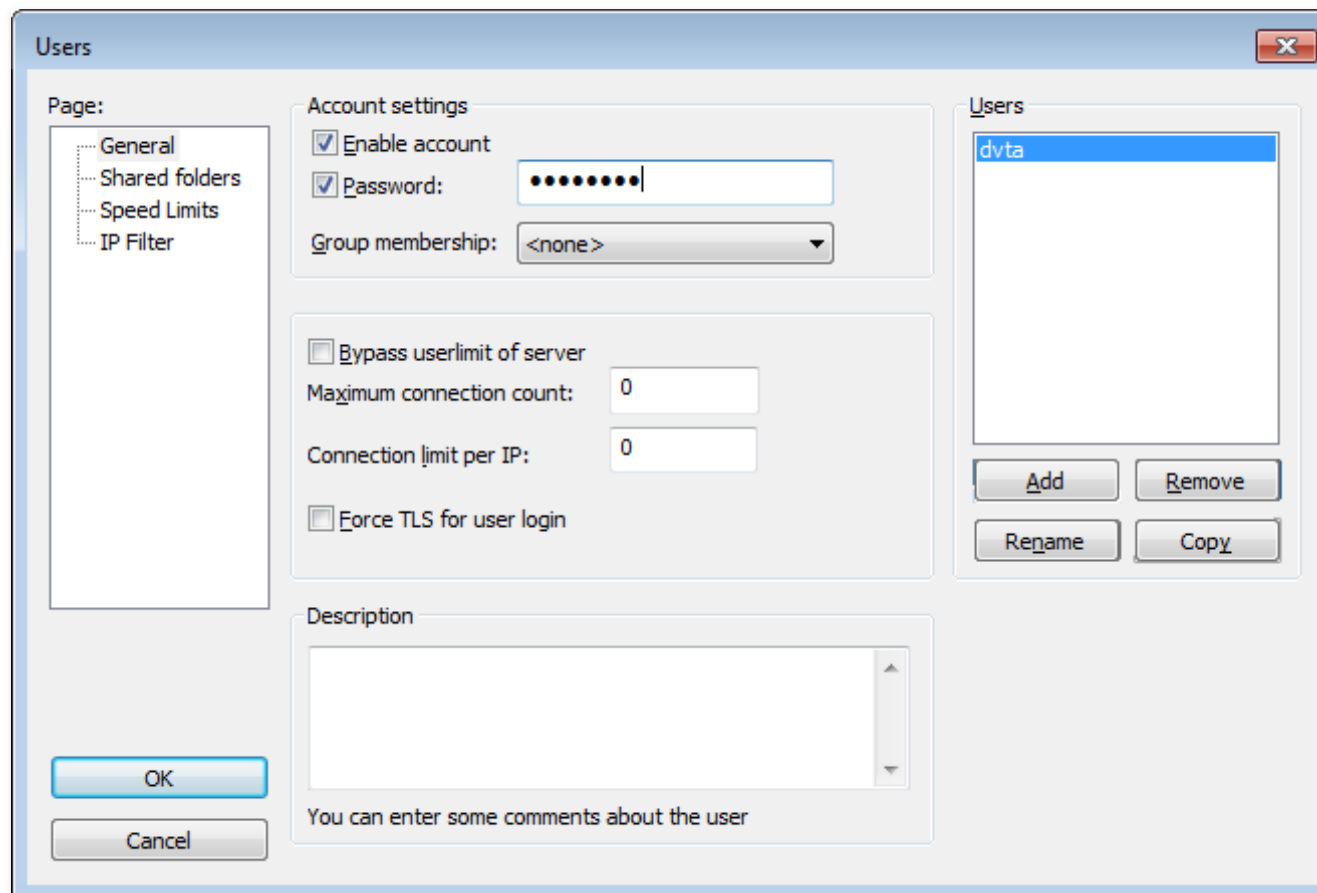


Restarting the service

5. Setup the FTP Server

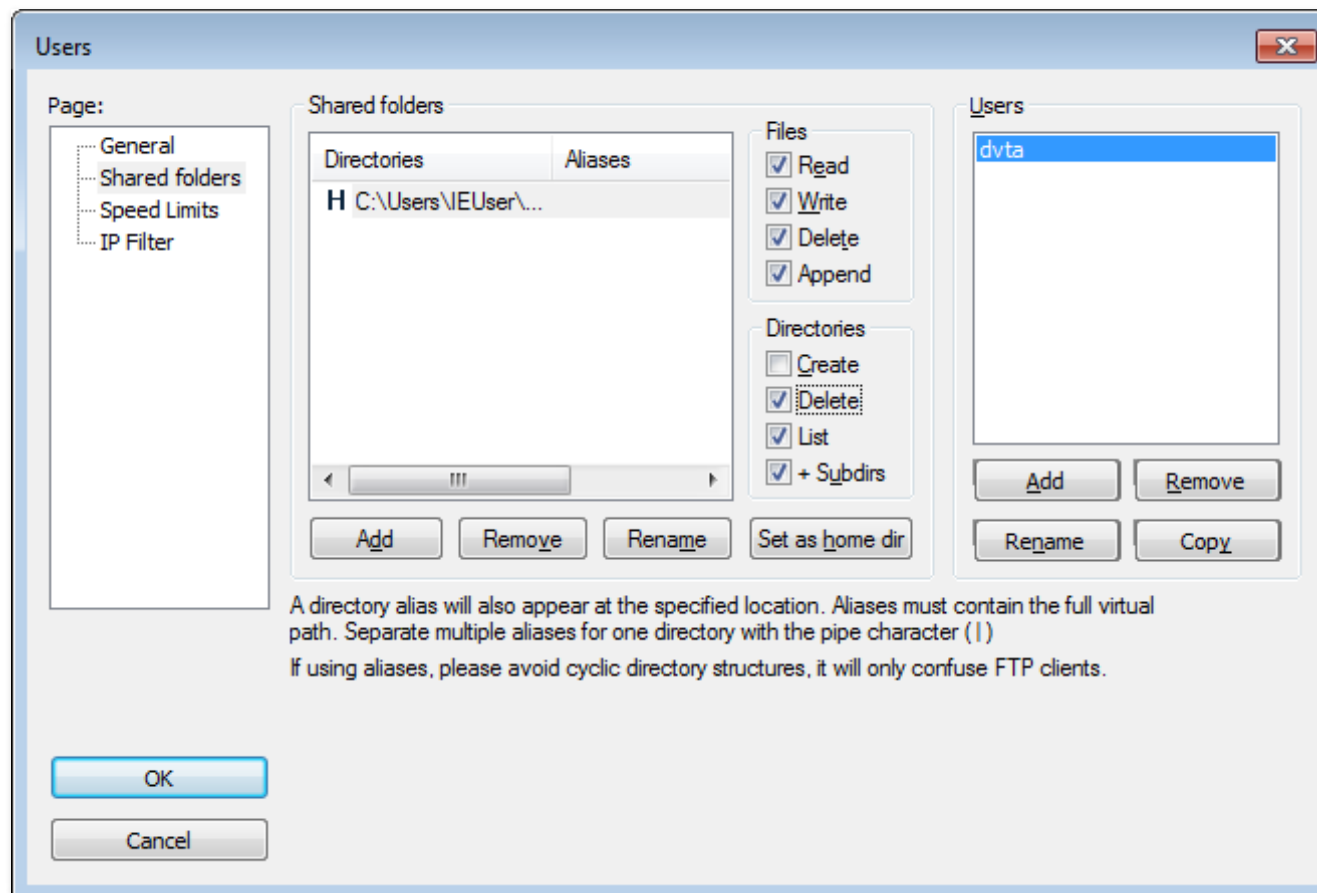
There's no need to install XAMPP. Manually install and use FileZilla FTP server.

- Create a directory (this will be the FTP root directory), I named it `dvta-ftp` and put in on desktop.
- Download and install the Filezilla FTP server (or any other server of your choice).
 - <https://filezilla-project.org/download.php?type=server0>
- Use `Edit (menu) > Users`
 - Under `General`, create a new user called `dvta` (no need to add it to a group). Then check the password checkbox and enter `p@ssw0rd`.



Creating the "dvta" user

- Click on `Shared folders`, add the FTP directory from before (`dvta-ftp`), and select ACL.



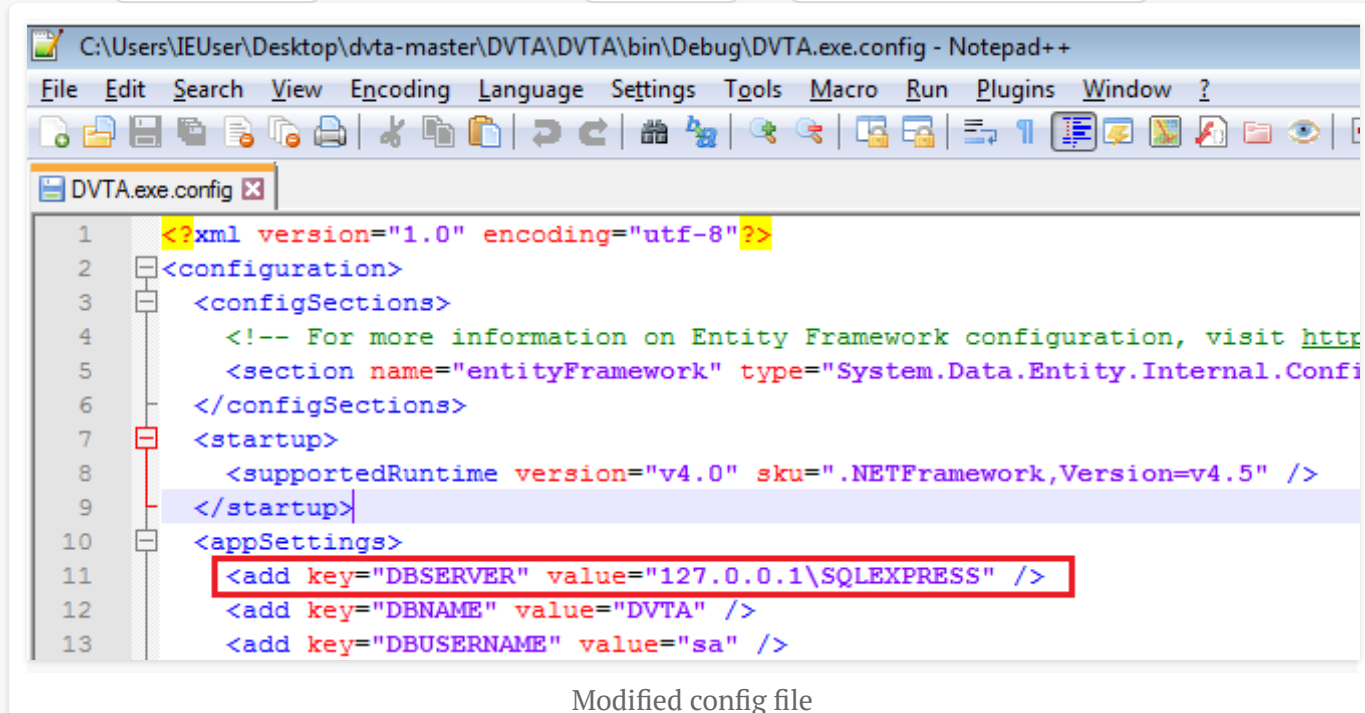
Giving access to the FTP user

Now our FTP server is ready and runs as a Windows service.

6. Modify DVTA to Connect to Our Local SQL Server

The binary is configured to look for the SQL and FTP servers at a hardcoded IP address. The SQL Server address is in the .NET config file (which is just an XML file).

- Open `dvta-master\DVTA\DVTA\bin\Debug\DVTA.exe.config` (by default extensions are hidden on Windows so the extension might not be visible).
 - Under `appSettings` change value of `DBSERVER` to `127.0.0.1\SQLEXPRESS`.



- Note: The `Release` version in this fork has extra protections (the login button is disabled by default). We will use the `Debug` version for testing the connection to our SQL Server. Be sure to do the same for the `Release` build later.
- Now we can login with any of the test users and also register new users.
- Notes:
 - The `Fetch Time` button will return an error regardless. I think it is the cert pinning protection that we need to bypass later.

7. Fix the FTP Connectivity

Admin can backup server files to an FTP server. But the FTP's address is hardcoded. It's

`192.168.56.110`. We can see this in the source code at `\dvta-master\DVTA\DVTA\Admin.cs`

(search for

`Upload("ftp://192.168.56.110", "dvta", "p@ssw0rd", @pathtodownload+"admin.csv");`). We

want to change it to localhost.

- We can fix it in different ways:
 1. Modify the source code and recompile the app. That involves installing Visual Studio and I don't wanna do that.
 2. Modify the binary with dnSpy.
 3. This is not the case here but if the application used a hostname, we could redirect using the `hosts` file. This is a common approach with real world software.

7.1 Use dnSpy to Modify the Hardcoded FTP Address

Let's assume we do not know the FTP address. That means we need to:

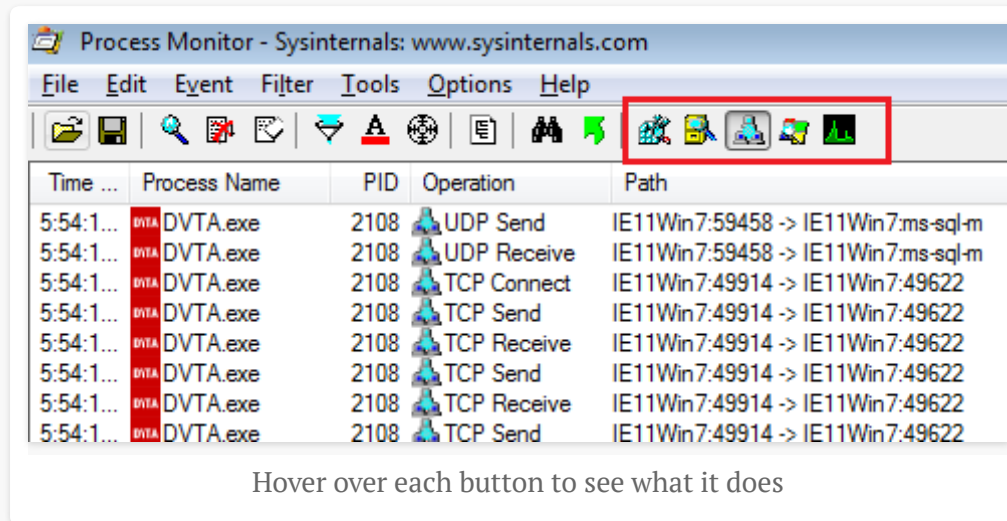
1. Discover the address.
2. Change the address in binary.

Discover the FTP Address

Use whatever method you are comfortable with. I used Procmon.

1. Start Procmon.
2. Run the application, login as admin and try to use the backup functionality.
3. Wait until you get the error message.

4. Set this filter in Procmon `Process Name is DVTA.exe`.
5. Remove all activities other than network by clicking on the buttons in the picture. Only keep the middle button enabled to display network activity.



6. ???
7. Profit².

Process Monitor - Sysinternals: www.sysinternals.com

File Edit Event Filter Tools Options Help

Time ...	Process Name	PID	Operation	Path
6:00:3...	DVTA.exe	2108	TCP Disconnect	IE11Win7:49915 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	UDP Send	IE11Win7:59458 -> IE11Win7:ms-sql-m
5:54:1...	DVTA.exe	2108	UDP Receive	IE11Win7:59458 -> IE11Win7:ms-sql-m
5:54:1...	DVTA.exe	2108	TCP Connect	IE11Win7:49914 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Send	IE11Win7:49914 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Receive	IE11Win7:49914 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Send	IE11Win7:49914 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Receive	IE11Win7:49914 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Send	IE11Win7:49914 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Receive	IE11Win7:49914 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Send	IE11Win7:49914 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Receive	IE11Win7:49914 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Send	IE11Win7:49914 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Receive	IE11Win7:49914 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Connect	IE11Win7:49915 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Send	IE11Win7:49915 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Receive	IE11Win7:49915 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Send	IE11Win7:49915 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Receive	IE11Win7:49915 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Send	IE11Win7:49915 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Receive	IE11Win7:49915 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Send	IE11Win7:49915 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Receive	IE11Win7:49915 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Send	IE11Win7:49915 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Receive	IE11Win7:49915 -> IE11Win7:49622
5:54:1...	DVTA.exe	2108	TCP Connect	IE11Win7:49916 -> 192.168.56.110:ftp
5:54:2...	DVTA.exe	2108	TCP Reconnect	IE11Win7:49916 -> 192.168.56.110:ftp

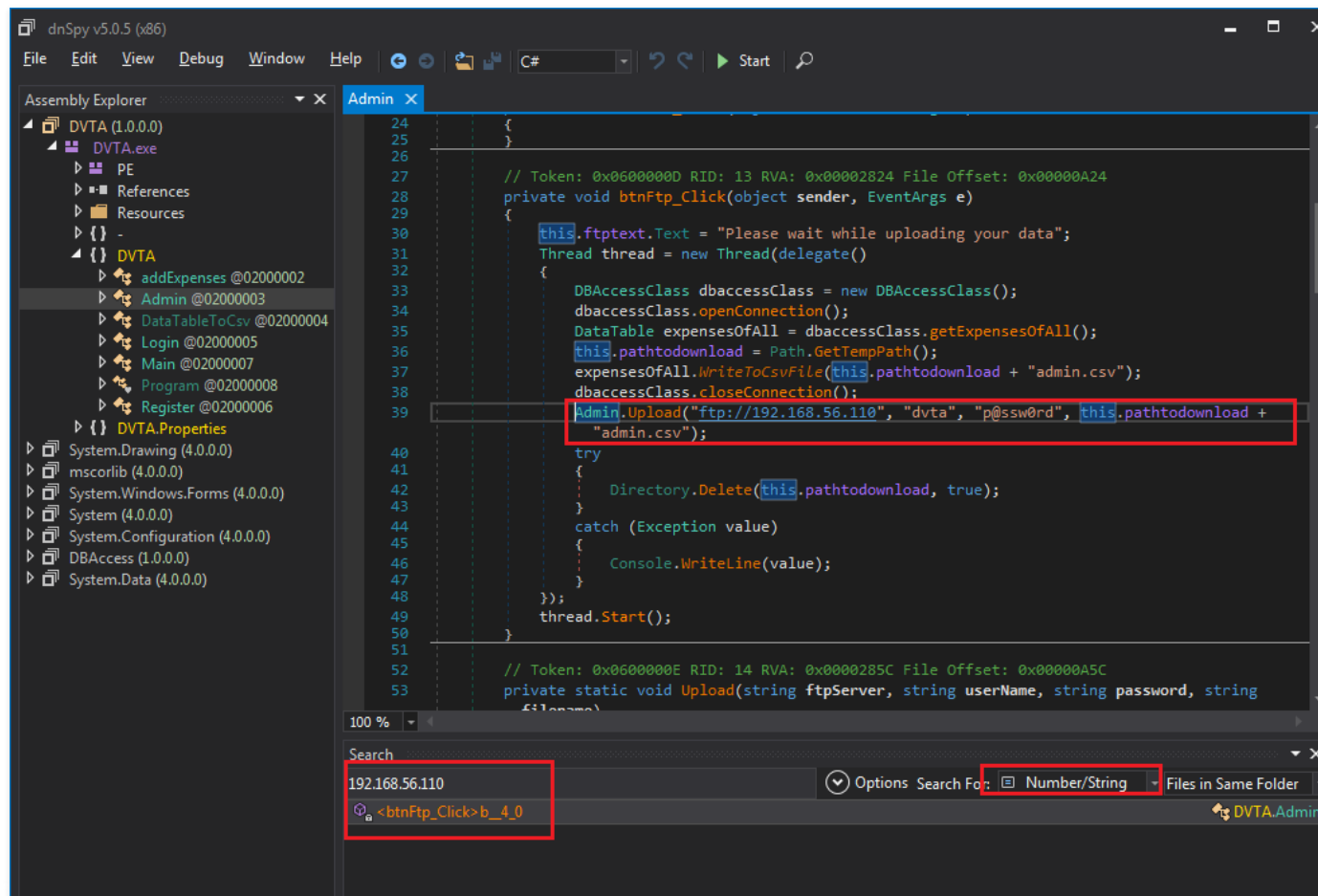
FTP address discovered

Modify the Address in Binary

Now we can use dnSpy to modify this address in the application.

- Create a backup of the original `dvta.exe`.
- Start dnSpy.
- Select `Edit (menu) > Search Assembly` and search for `192.168.56.110`. Choose `Number/String` for the `Search For` combo box. `All of the Above` does not search for text (unfortunately).

- Click on the search result and Voila! We have our FTP address (and password).



FTP address in code

- Right-click and select `Edit Method`. Now we can edit the C# source code.
 - Now listen kids. Back in my day we didn't have such nice things, we had to hand-craft CIL instructions walking uphill in the snow.

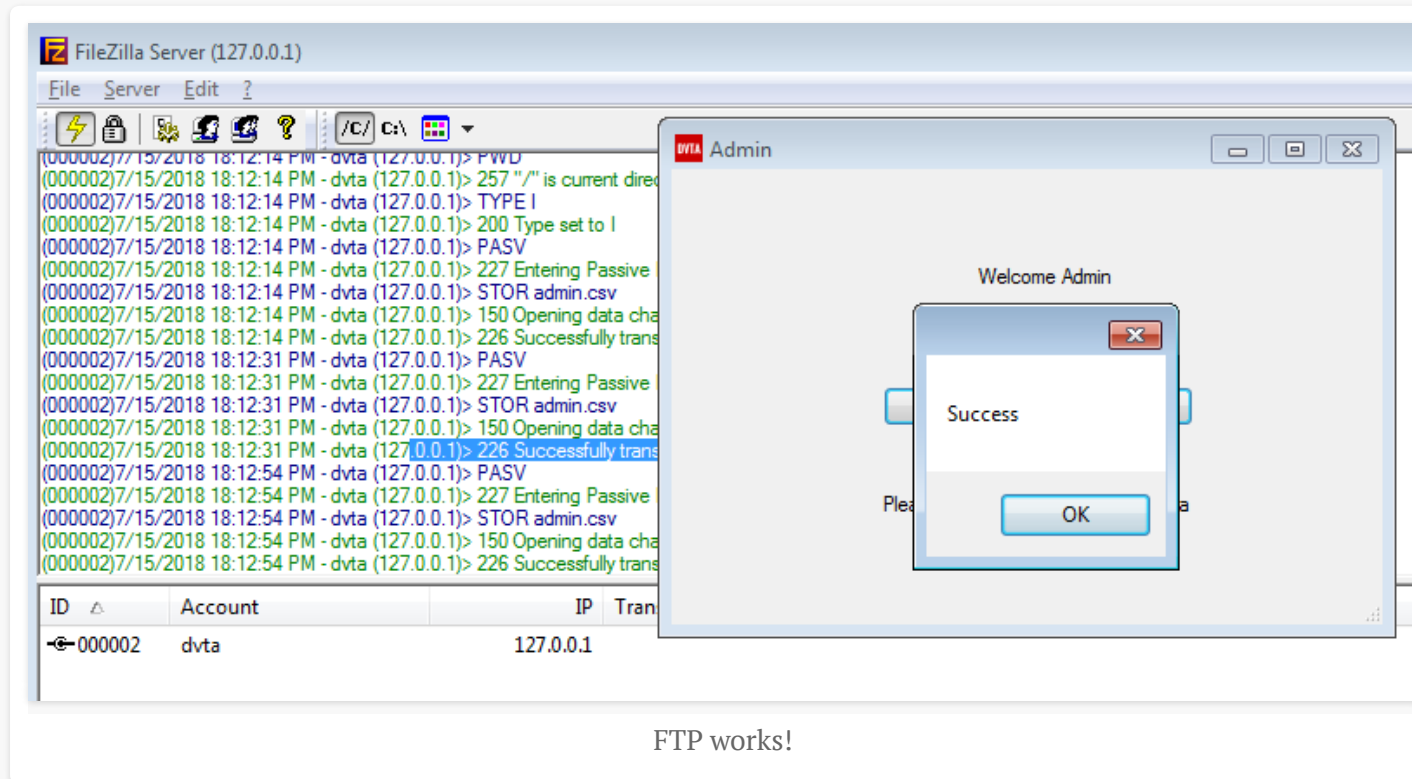
- Modify `192.168.56.110` to `127.0.0.1`.

```
7 using DBAccess;
8
9 namespace DVTA
10 {
11     // Token: 0x02000003 RID: 3
12     public partial class Admin : Form
13     {
14         // Token: 0x0600000D RID: 13 RVA: 0x00002824 File Offset: 0x00000A24
15         private void btnFtp_Click(object sender, EventArgs e)
16         {
17             this.ftptext.Text = "Please wait while uploading your data";
18             Thread thread = new Thread(delegate()
19             {
20                 DBAccessClass dbaccessClass = new DBAccessClass();
21                 dbaccessClass.openConnection();
22                 DataTable expensesOfAll = dbaccessClass.getExpensesOfAll();
23                 this.pathtodownload = Path.GetTempPath();
24                 expensesOfAll.WriteToCsvFile(this.pathtodownload + "admin.csv");
25                 dbaccessClass.closeConnection();
26                 Admin.Upload("ftp://127.0.0.1", "dvta", "p@ssw0rd", this.pathtodownload + "admin.csv");
27             });
28             try
29             {
30                 Directory.Delete(this.pathtodownload, true);
31             }
32             catch (Exception value)
33             {
34                 Console.WriteLine(value);
35             }
36         }
37     }
38 }
```

Modified FTP address

- Click on `Compile` and now the code has changed **but it's not saved to any file yet**.
- Select `File(menu) > Save Module` to save the executable.

- Now you can run the patched binary and use the FTP functionality.



Conclusion

We setup DVTA in a VM and patched it to connect to our local FTP server. Now things are ready to go and we can start hacking the application. In the next post I will start working on the application.

1. I did not read the solution because I wanted to do things my own way and learn. [\[return\]](#)

2. I am not sure why the application is trying to do reconnect instead of normal [TCP Connect](#)
.[\[return\]](#)

Posted by Parsia • Jul 15, 2018 • Tags: [dnSpy](#)

[Istanbul Tips and Tricks](#)

[DVTA - Part 2 - Cert Pinning and Login Button](#)

0 Comments

Parsiya

[1 Login](#) ▾

[Recommend](#)

[Tweet](#)

[Share](#)

Sort by Best ▾



Start the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS [?](#)



Name

Be the first to comment.

[Subscribe](#)

[Add Disqus to your site](#)

[Disqus' Privacy Policy](#)

DISQUS

Copyright © 2019 Parsia - [License](#) - Powered by [Hugo](#) and [Hugo-Octopress](#) theme.

