Faith Dev Guide

-Johnz

# Latest update: John.Zhang, 2015-02-15

# I, Dev Box Chef Setup

This process will install all the needed libraries and applications (.NET 2/3/4, MVC 2/3/4, Gallio, MongoDB, Visual Studio 2013) needed to work with the main Fellowship One applications (Portal, InFellowship and the API’s). This will also setup the local file shares used for people images, InFellowship branding, etc. This will also install a local instance of SQL server with the supporting databases. It will setup all of the IIS websites, hostnames and app pools along with changing the hosts file to add the local aliases.

This chef recipe is assuming it’s being run on a fresh install of Windows 8.1:

## Install Win8.1 (IT will do)

If you want to have more than 1 driver, please make sure driver C has about 150G.

[\\wsfscn02\ISO\SW\_DVD9\_SA\_Win\_Ent\_8.1\_64BIT\_English\_-2\_MLF\_X19-49847.ISO](file:///\\wsfscn02\ISO\SW_DVD9_SA_Win_Ent_8.1_64BIT_English_-2_MLF_X19-49847.ISO)

## Get on the Active Domain (IT will do)

**For below steps, maybe West VPN is needed to download files.**

## Install Chef Client

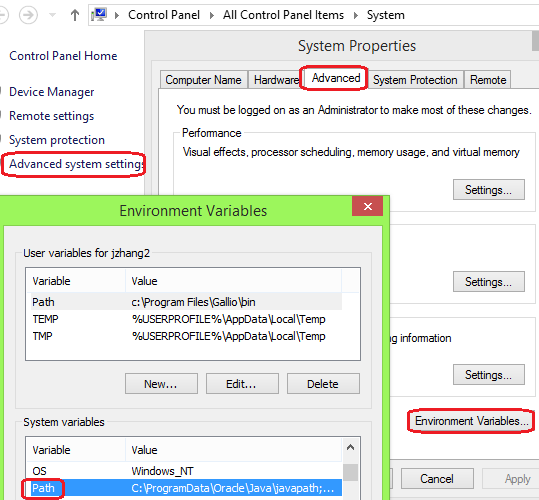
[\\wsfscn01\Faith Dev Xian\Faith\F1 Development Box Setup\dev-chef-installs](file:///\\wsfscn01\Faith%20Dev%20Xian\Faith\F1%20Development%20Box%20Setup\dev-chef-installs)\chef-client-11.6.0-1.windows.msi (This one will be better to work with current “cookbooks”, the latest version chef-client will meet some problems with current “cookbooks”.)

## Install Git

<http://git-scm.com/download/win>

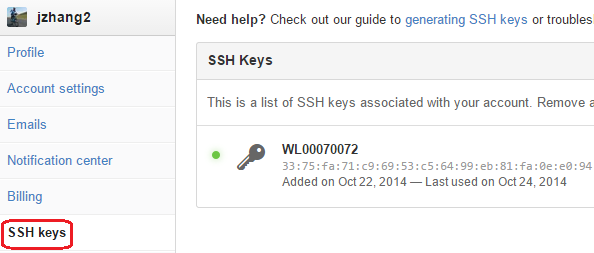
* 1. Register an account in “<https://github.com>”, and then send it to Josh (Josh.Berryhill@activenetwork.com) who will add you into the F1 Git groups.
  2. Make sure add Git bin folder to the Windows Path, otherwise the recipe won’t work

C:\Program Files (x86)\Git\bin;



* 1. Generate [ssh key](https://help.github.com/articles/generating-ssh-keys) and upload to the [github.com](https://github.com/settings/ssh)

<https://help.github.com/articles/generating-ssh-keys>

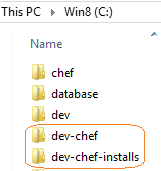


If you meet error “Could not open a connection to your authentication agent” when you generate “ssh key”, please just execute “ssh-agent bash” in git bash.

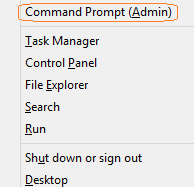
## Extract the “dev-chef.zip” folder into root c:\, and copy “dev-chef-installs” into root c:\

Get all F1 dev tools from “[\\wsfscn01\Faith Dev Xian\F1 Development Box Setup](file:///\\wsfscn01\Faith%20Dev%20Xian\F1%20Development%20Box%20Setup)”

Or get from “[\\active.local\data\US-TX-Dallas\Development\Development Box Setup](file:///\\active.local\data\US-TX-Dallas\Development\Development%20Box%20Setup)” that you need to enter SN for accessing share folder.



## Run command prompt as Administrator



* 1. Navigate to “c:\dev-chef”
  2. You need the Win8.1 install disc on the “d:\” drive
  3. **Run “install”** (install.cmd),

This will take more than hour, During the installing, maybe prompt for entering “passphrase” is required that you used to create ‘[ssh key](https://help.github.com/articles/generating-ssh-keys)”.

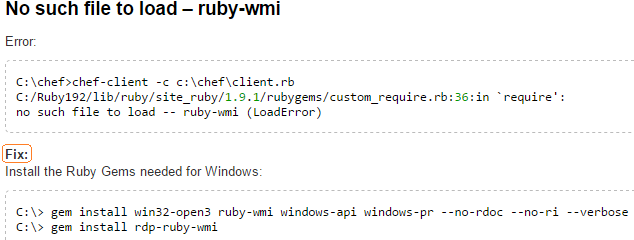
Two problems maybe pop up:

* + 1. SSL validation of HTTPS requests is disabled

Add “ssl\_verify\_mode :verify\_peer” in the end of “C:\dev-chef\config\ solo.rb” file.

* + 1. Cannot load such file -- ruby-wmi

[https://wiki.opscode.com/display/chef/Common+Errors#CommonErrors-Nosuchfiletoload–ruby-wmi](https://wiki.opscode.com/display/chef/Common+Errors%23CommonErrors-Nosuchfiletoload–ruby-wmi)



## After the install, go to edit the local users (web\_user, svc\_user, \*\_aspnet), so that their passwords don’t expire

## After the install done, you will find F1 codes in “C:\dev”

## Command Prompt as Administrator, then to run “C:\dev\databas\Local\_-\_UpdateAll.cmd”

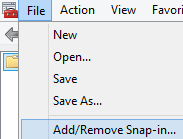
## Extract the db backup files to “c:\datebase”, and restore db to your local SQL Server

## Execute “c:\dev\shared\localresources\scripts\3\_create\_logins\_local.sql”

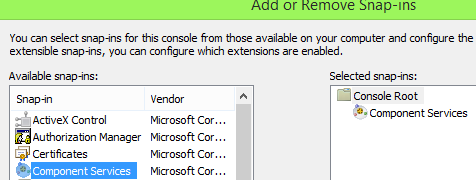
* 1. It create users that used in code
  2. There will be errors that you can ignore

## Run “Microsoft Management Console (mmc.exe)”

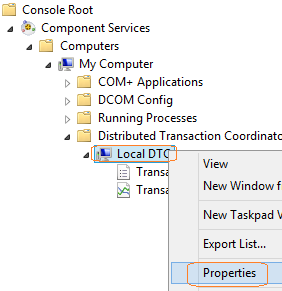
* 1. File-> Add/Remove Snap-In…



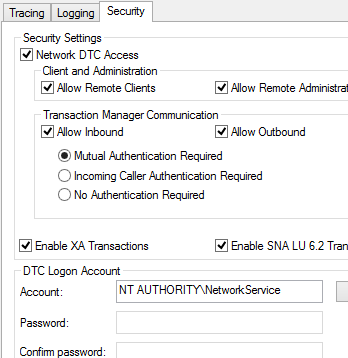
* 1. Add Component Services



* 1. Open Properties for “Console Root -> Component Services -> Computers -> My Computer -> Distributed Transaction Coordinator -> Local DTC”

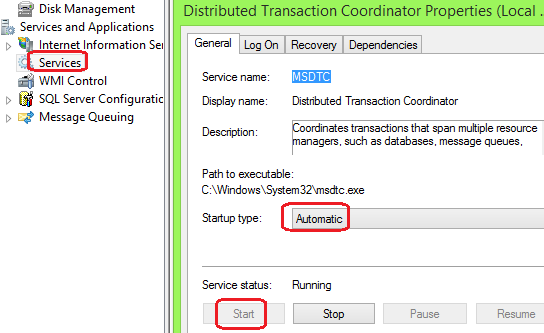


* 1. From “Security” tab, check all checkboxes and then hit Apply



* 1. **Distributed Transaction Coordinator**

Update it to run Automatic, as infellowship verify account register will use this service



## Install VS2013

[\\wsfscn01\ISO\Visual\_studio\en\_visual\_studio\_professional\_2013\_x86\_dvd\_3175298.iso](file:///\\wsfscn01\ISO\Visual_studio\en_visual_studio_professional_2013_x86_dvd_3175298.iso)

This also need enter SN, then IT will make license active

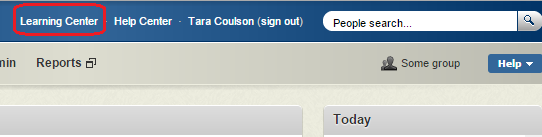
## Windows updates failed, please connect West VPN, then have a try

# II, F1 Part

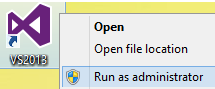
## F1 New guys learning center

<https://portal.fellowshipone.com/>

Account/PW: tcoulson **/**FT.Admin1, Church code: DC



## VS2013 Run as administrator



* 1. Otherwise, you will meet the error cannot load infellowship project

**2.2** Nuget Error:

Need connect VPN, as China blocks the site: <http://www.nuget.org/>

## [Git learning](https://www.atlassian.com/git/tutorials/setting-up-a-repository), works with code (“C:\dev\\*”):

<https://www.atlassian.com/git/tutorials/setting-up-a-repository>

<http://git-scm.com/>

1. *Git config*:

<http://git-scm.com/book/en/v2/Getting-Started-First-Time-Git-Setup>

Run “Git Bash”

$ git config --global user.name "John Zhang"

$ git config --global user.email [John.Zhang@ActiveNetwork.com](mailto:John.Zhang@ActiveNetwork.com)

$ git config --list

1. Getting a Git Repository

$ git clone <https://github.com/jzhang2/GitTest> MyGitTest

1. Check in Updates

$ git add file.txt ‘stage file.txt

$ git diff ‘show updates

$ git status

$ git commit –m “update file.txt” ‘commit updates to local, not git serber

$ git push ‘check in to server

1. Show Logs

$ git log ‘you can use Space btn to scroll, and q to exit

1. View Old Version

$ git checkout <commit> <file>

$ git checkout master ‘get back to “current” state of project

$ git checkout HEAD <file> ‘checkout the latest version project

1. Undoing Changes

$ git revert ‘undo a public commit

$ git reset ‘undo local changes for tracked files

$ git reset --hard ‘make your local directory match to recent version commit

$ git clean –df ‘remove un tracked directory

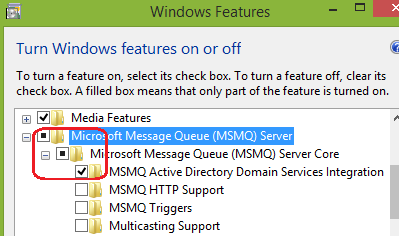
1. Branch and Merge

$ git branch

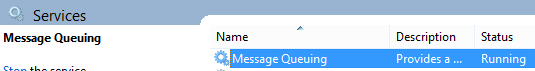
1. And so on

## MSMQ

<http://technet.microsoft.com/en-us/library/bb684791.aspx>

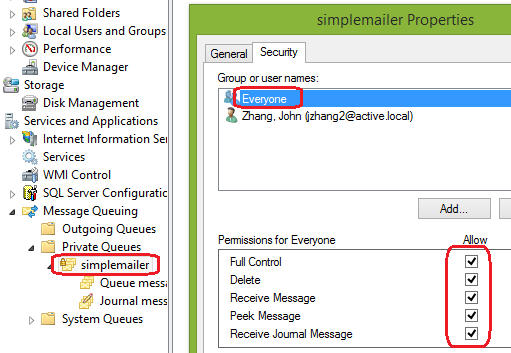


Then goto “Services” control panel, make sure “Message Queuing” is running:



From debug codes, you will find F1 used MSMQ that named: “SimpleMailer”,

So to create “simplemailer” by yourself, and give full control to “everyone”:



## Scrum Tool

<http://scrumworks.dev.activenetwork.com/scrumworks/taskboard>

Chat with our NA manager - Josh, to let he add you into it, and then you can login Scrum.

For Xi’an team, the scrum team is – “**FAITH-Nice East**”.

## F1 Codes and Business, church code: DC

* 1. <http://dc.infellowship.local>
  2. **Some existing user**

kklein@fellowshiptech.com /udliskek

ft.autotester@gmail.com /FT4life!

mlindsley@fellowshiptech.com / John3:16

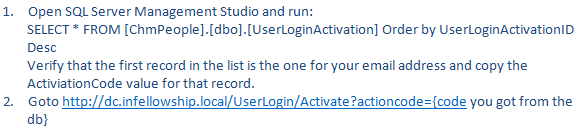
* 1. **Create and Verify a new user**

Create a new user, then F1 will send a verify email to you that saved in

“C:\Windows\System32\msmq\storage\r\*\*\*.mq”

Please get the verify URL from above file, and then open it and fill verify info.

Or use below way is Ok:



* 1. <http://portal.local>

ft.tester/FT4life!

## MVC learning:

<http://www.asp.net/mvc/overview/getting-started/introduction/getting-started>

# III, F1 API Part (Mazelin, Tracy)

Take a look at the API documentation here: <http://developer.fellowshipone.com/docs/>

 I’m not sure what type of monitoring you are working on but we used to use an app called New Relic.  It was set up with some simple ping requests to each of our API realms via:

People: [https://dc.fellowshiponeapi.com/v1/Util/Docs.help](https://dc.staging.fellowshiponeapi.com/v1/Util/Docs.help)

Giving: [https://dc.fellowshiponeapi.com/giving/v1/Util/Docs.help](https://dc.staging.fellowshiponeapi.com/giving/v1/Util/Docs.help)

Groups: [https://dc.fellowshiponeapi.com/groups/v1/Util/Docs.help](https://dc.staging.fellowshiponeapi.com/groups/v1/Util/Docs.help)

Events: [https://dc.fellowshiponeapi.com/events/v1/Util/Docs.help](https://dc.staging.fellowshiponeapi.com/groups/v1/Util/Docs.help)

# IV, ServiceU Part

## ServiceU Learning Vidoes

[http://www.serviceu.com/support/ServiceU%20Online%20Help/Default.htm#ServiceU Videos.htm](http://www.serviceu.com/support/ServiceU%20Online%20Help/Default.htm%23ServiceU%20Videos.htm)

<https://devmy.serviceu.com/>   (Maybe need VPN from China to access)

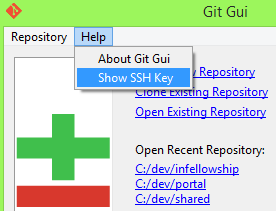
## Access ServiceU git code lab

<https://memgit.serviceu.com/serviceu/serviceu>

Sure, gitlab account is needed, and get access permission of SU repo from Josh

* 1. **Add SSH Key for ServiceU git lab, then you can download codes to your machine:**

<https://memgit.serviceu.com/profile/keys>



* 1. **Work with codes:**
  2. **Get codes to local machine**:

“C:\dev”, then open “Git Bash”

$ git clone [git@memgit.serviceu.com:serviceu/serviceu.git](mailto:git@memgit.serviceu.com:serviceu/serviceu.git)

$ git clone git@memgit.serviceu.com:serviceu/hpp.git

## Dev setup

### **Before run this part, please save your “hosts” file in** “C:\Windows\System32\drivers\etc\”

As SU script will re-write the file context, and after SU setup successfully, please copy context of original “hosts”, and paste them in the end of new “hosts” file.

### Developer Setup

<https://memgit.serviceu.com/serviceu/docs/blob/master/developer-setup.md>

* + 1. “**Windows Features**” is required. Copy out the script, and change to show in one line, then run in powershell.
    2. For “**Utilitites**”, “**Source Control**”, and “**Programming Tools**” just pick you liked tools to install will be Ok, it means don’t need to install the all the doc mentioned.
    3. For “**Source Setup**”, after pulled down coeds, and add the PATH, then turn to run “[Chef Setup](https://memgit.serviceu.com/serviceu/chef-repo/blob/master/docs/install-dev.md)”, after “Chef Setup” done, then turn to run “start-build.bat”

### Chef Setup

[https://memgit.serviceu.com/serviceu/chef-repo/blob/master/docs/install-dev.md](https://memgit.serviceu.com/serviceu/chef-repo/blob/master/docs/install-dev.md#knife-solo-passwords-config)

* + 1. **Install chef solo (**powershell run as admin**)**

“secret” is a file without extend name, and fill any string you think is Ok.

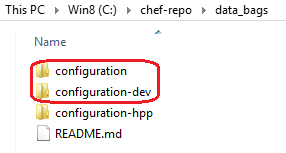


* + 1. **Knife Solo \* (cmd)**

wildcard\_password: Launch$$990704

magensa\_cert\_password: ServU@2012

Two created files will saved in “C:\chef-repo\data\_bags\...”

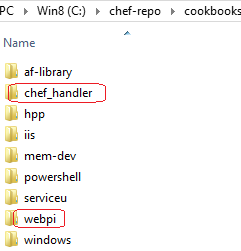


* + 1. **Run Chef (**c:\chef-repo\docs\install-dev.bat**)**

1. **“C:\chef-repo\.chef\ solo.rb”**

Add “ssl\_verify\_mode :verify\_peer” in the end of the file, to fix the warning ssl validation

1. **Copy “chef\_handler” and “webpi”** from F1 cookbooks-“ C:\dev-chef\cookbooks” to “C:\chef-repo\cookbooks”:



1. **Add below lines** in the end of “C:\chef-repo\cookbooks\mem-dev\metadata.rb” file:

depends "af-library"

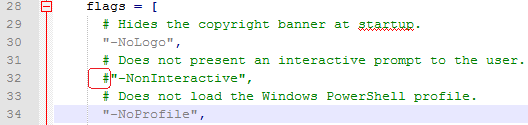
depends "windows"

depends "powershell"

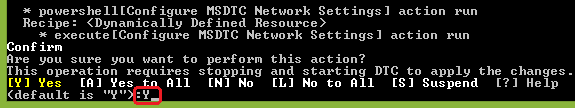
depends "iis"

1. **Update “default.rb” in C:\chef-repo\cookbooks\powershell\providers\:**

Use “#” to commit “-NonInteractive”,

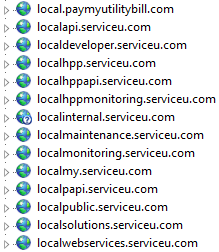


Then during the running of script, it will prompt as below, then you need enter “Y”

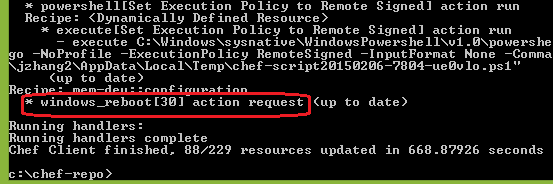


For this part script changes, as I meet the problem – powershell to set DTC setting always failed with the original coding.

1. **If you run “c:\chef-repo\docs\install-dev.bat” failed ago, then please remove all SU sites in IIS, and before you run the script again**



1. **Manual Reboot computer is required:**



Then run the script again.

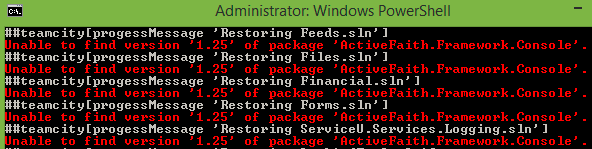
## Source Setup – running “build\start-build.bat” in CMD with admin permission

Or run “./start-build.ps1” in PowerShell with admin permission is Ok

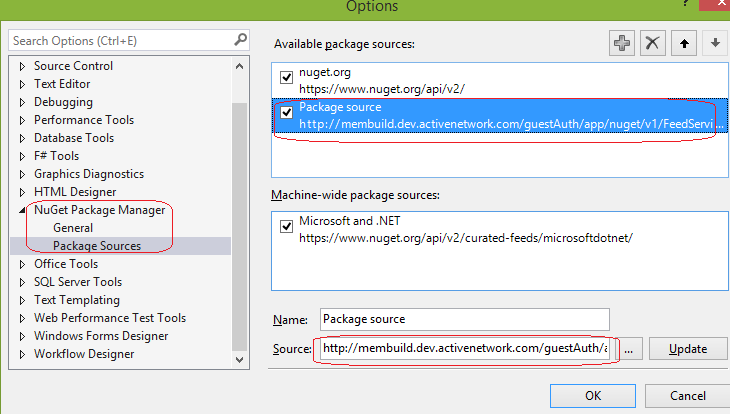


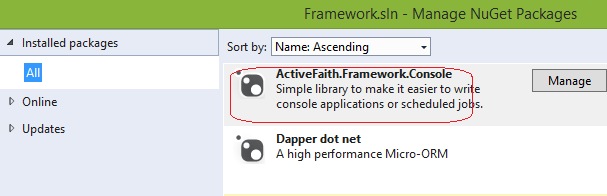
**Some problem that I meet during configuration:**

1. **Miss “ActiveFaith.Framework.Console”**

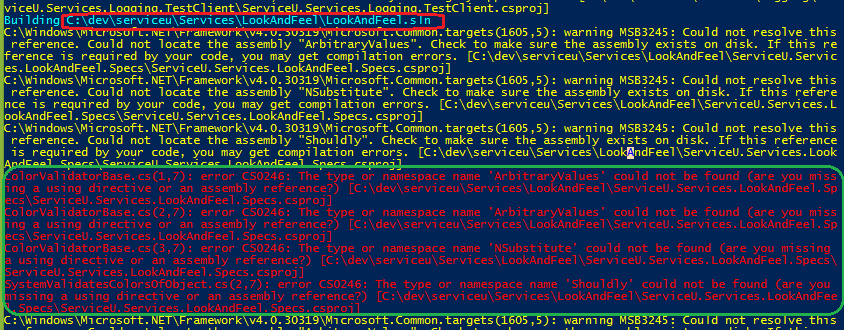


Open one solution, such as “C:\dev\serviceu\Framework\Framework.sln” in VS, then add “<http://membuild.dev.activenetwork.com/guestAuth/app/nuget/v1/FeedService.svc/>” in “NuGet Package Manager – Package Sources”, otherwise you will meet compile errors as missed “ActiveFaith.Framework.Console” errors.

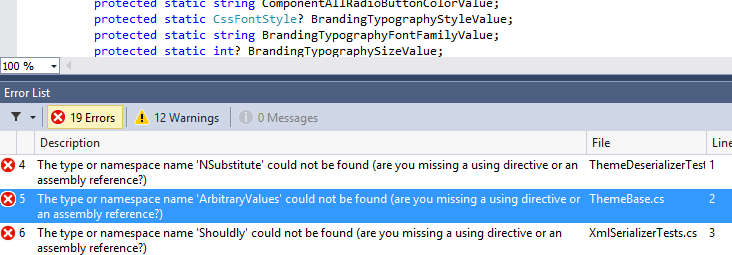




1. **Miss some reference in project**



Open the project in VS by manual, then try to build it, you will meet below error (reference cannot work):

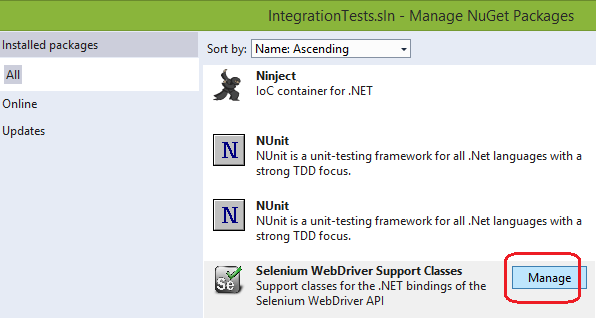


Just open “TOOLS->…->Manager Nuget Packages for Solution…”



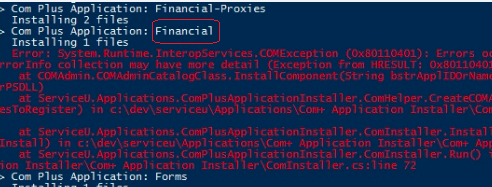
Then try rebuild the project again, you will find it successfully.

Otherwise, you need to go to “Manage NuGet Packages”, and for special missed reference, click the “Manager” button, then close all sub forms, and try re-build the project again.



**For fixing reference error for all projects, you need to do this step many times; maybe sometimes you need VPN, until the script shows “Building Successfully” prompts.**

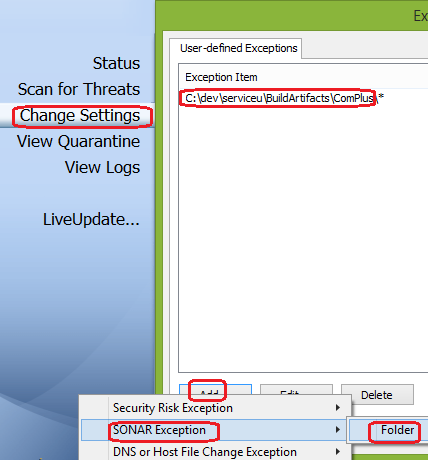
1. **COM plus error**



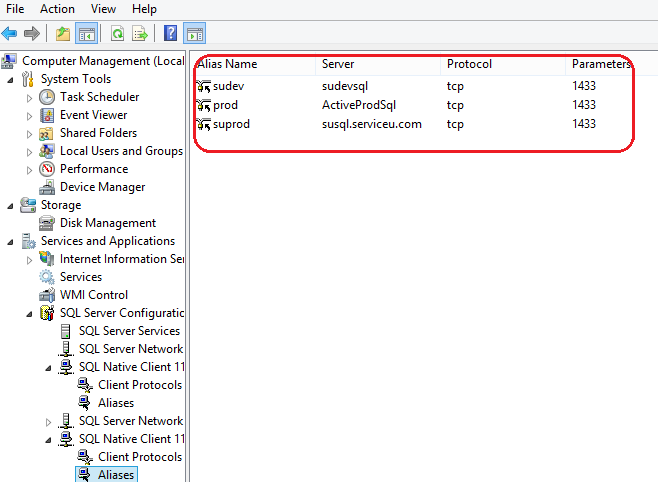
Go to “Control Panel\All Control Panel Items\Administrative Tools\Component Services”, and delete error COM+

**Solution**: just add the folder “”C:\dev\serviceu\BuildArtifacts\ComPlus” in “Symantec >> Change Settings >> Exceptions>> Configure Settings”

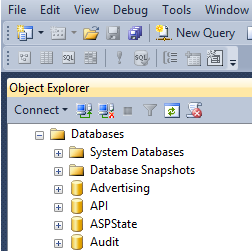
[If you haven’t the permission, please enter Service-Now to get.]



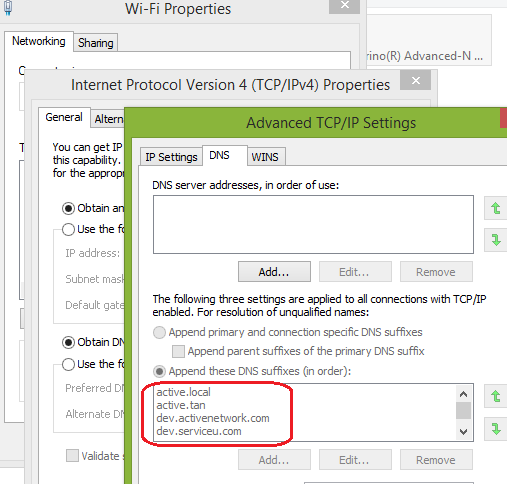
## ServiceU DB (server: sudevsql, ip: 10.119.33.251)



Then you can connect SU db



## We should have the same DNS suffixes with NA members’ now:



# V, Some helpful Tools: