

# .Net 项目 Jenkins 持续集成实践

---

title: Continuous integration Guide

## 目录

安装

插件管理

权限管理

持续集成步骤分解

端到端持续集成真实演练

Auto-deployment-with-docker-and-ansible

演示录像


一些项目实践经验

Troubleshooting

**作者说明**

GitChat

Download **Jenkins 2.73.1 for windows** from <https://jenkins.io> will automatically install Jenkins as window service.

Downloads > jenkins-2.73.1.zip			
Name	Type	Compressed size	Password
 jenkins.msi	Windows Installer Package	115,991 KB	No

当你解压下载的zip后，里面是一个名为jenkins.msi的可执行文件，双击安装，一路默认选项到最终完成。

此时浏览器会自动打开<http://localhost:8080/>。

会有一些交互的页面需要你去执行一些初始化的操作，特别是在定制化插件的步骤，会让你自主选择需要安装的部分插件。勾选上一些.NET 项目所需的插件。

#### Getting Started

# GitChat

## Getting Started

<input type="radio"/> Subversion Plug-in	<input type="radio"/> MSBuild Plugin	<input type="radio"/> HTML Publisher plugin	<input type="radio"/> Build Pipeline Plugin
<input type="radio"/> GitHub plugin			

Note that the full list of plugins is not shown here. Additional plugins can be installed in the

- HTML Publisher plugin

## Create Admin

上述部分插件安装完成后，界面会提示创建Admin账号。示例信息如下：

Username: admin

Password: admin

Fullname: admin

Email: liuning0820\@outlook.com

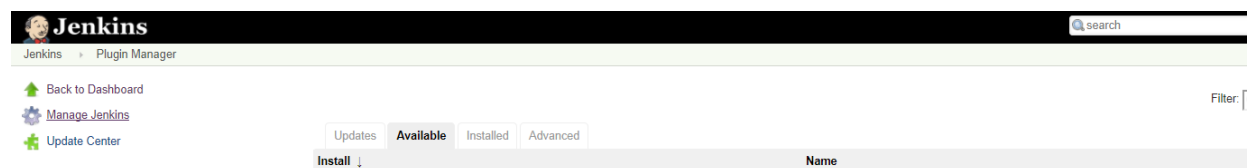
## Install Git Tool

Download the git SCM tool from <https://git-scm.com/download/win> and install it on the Jenkins server.

## 插件管理

### 安装 MStest Plug-In

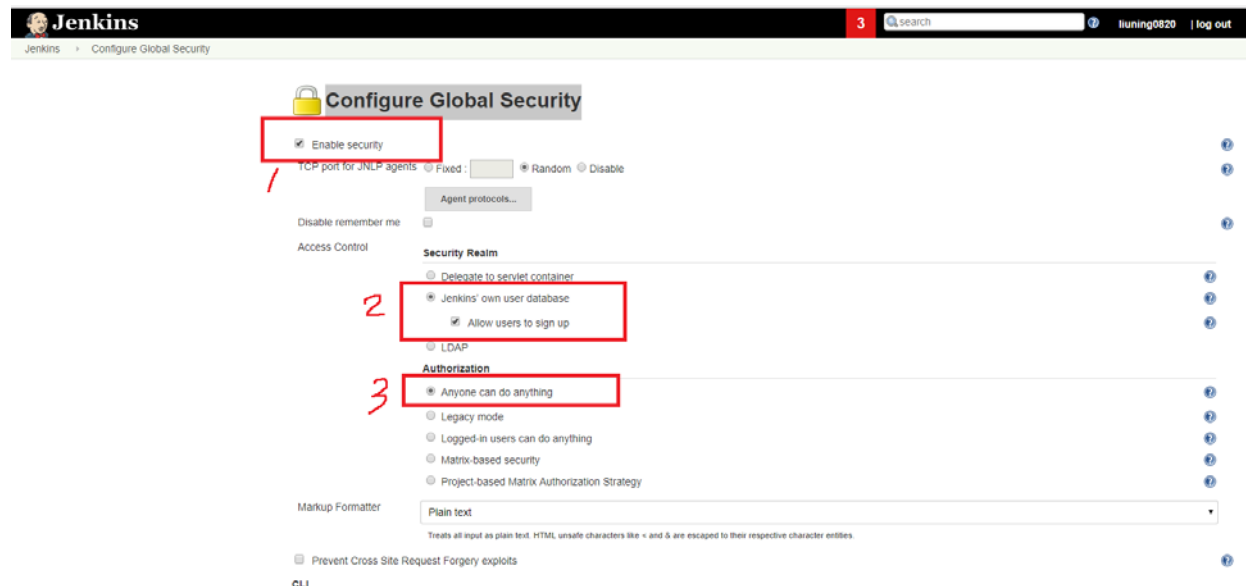
Click “Manage Jenkins” link, click “Manage Plugins” link, Click “Available” tab and type the name of the desired plugin in the filter text box. Install the “MSBuild Plugin” – This plugin converts MStest TRX test reports into html.



# 权限管理SecurityManagement

## Enable Security

### Manage Jenkins-> Configure Global Security

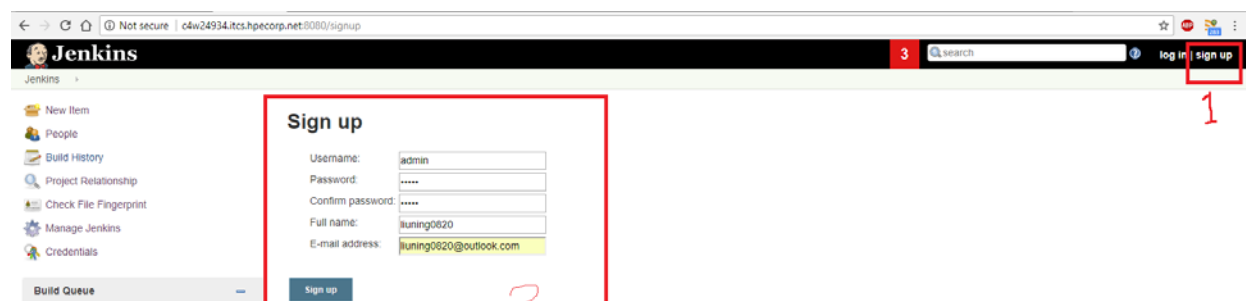


The screenshot shows the Jenkins 'Configure Global Security' page. It includes sections for enabling security, setting the security realm, and authorization. Red boxes and numbers highlight the following steps:

- 1. **Enable security**: The checkbox is checked.
- 2. **Security Realm**: 'Jenkins' own user database' is selected, and 'Allow users to sign up' is checked.
- 3. **Authorization**: 'Anyone can do anything' is selected.

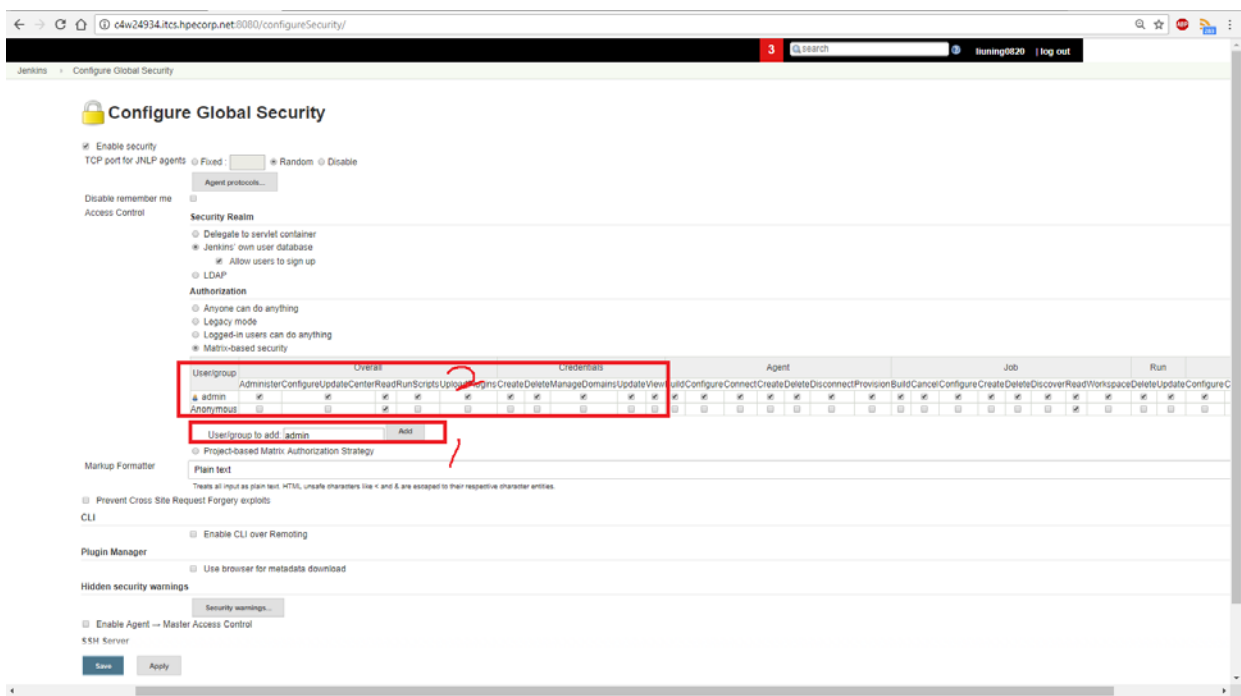
**Note:** In this step, we firstly set the authorization to “Anyone can do anything”. 因为到这一步，你还没有创建任何用户，需要去创建一些用户后，然后选择Matrix-based security。

## Signup the “admin” user



The screenshot shows the Jenkins 'Sign up' page. A red box highlights the sign-up form, and a red box with the number '1' highlights the 'sign up' button in the top right corner. The form contains the following fields:

- Username: admin
- Password: [masked]
- Confirm password: [masked]
- Full name: liuning0820
- E-mail address: liuning0820@outlook.com



## 持续集成步骤分解

Integrate Jenkins with GitHub

Install “GitHub plugin” if it is not installed yet:

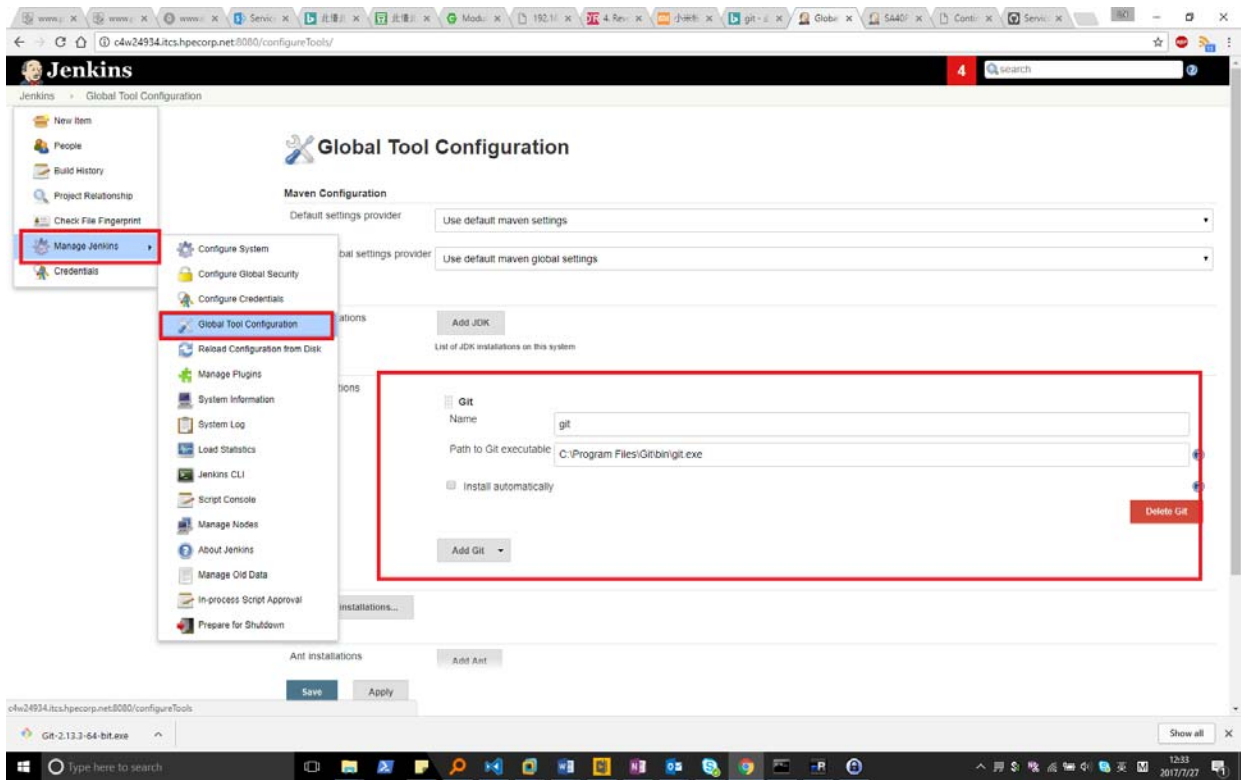
Open the Plugin Manager page.

Search “GitHub plugin” in the “Available” Tab and Install it.

Install git scm tool in the Jenkin server:

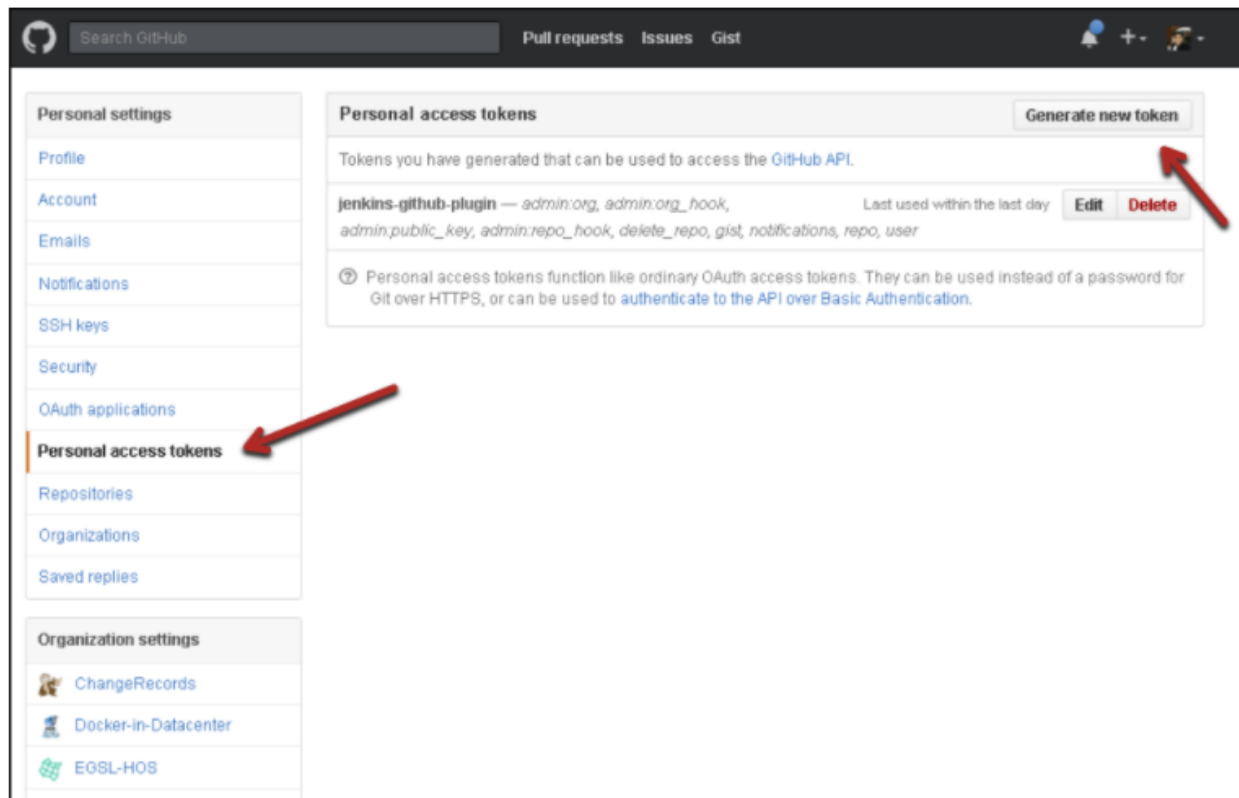
Download git <https://git-scm.com/> and install it in the Jenkins server, remember the installation path of git.exe. By default it is “C:\Program Files\Git\bin\git.exe”

Then configure the git Path on Jenkins below

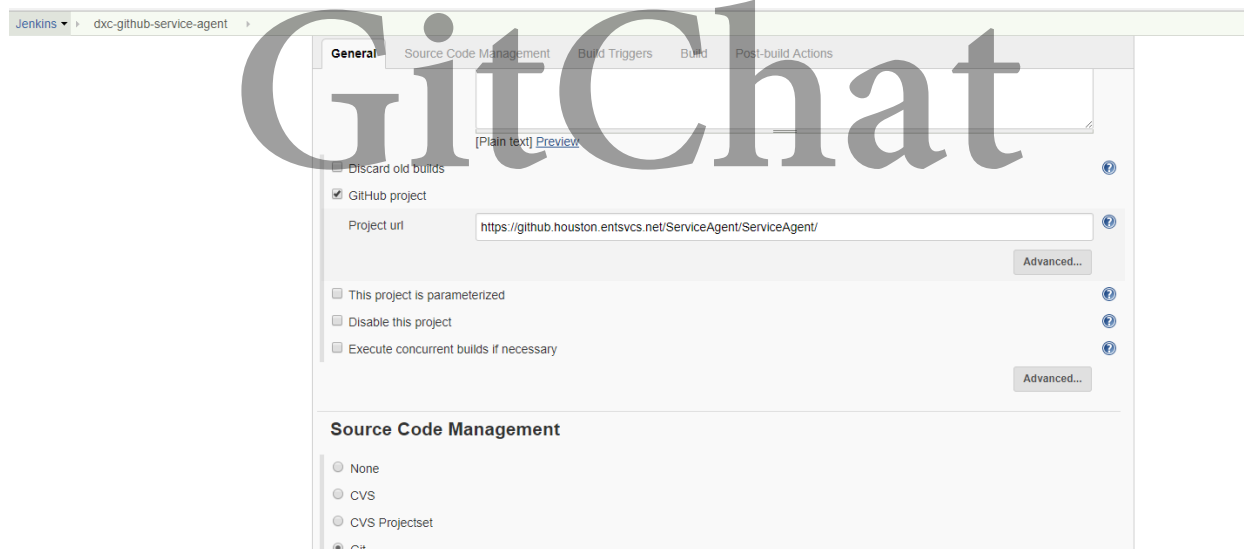


## Add Credentials to Access GitHub





Then create a job the fetch the source code.



And configure the path of the MSBuild.

**MSBuild**

MSBuild installations

MSBuild Name: dot-framework-msbuild

Path to MSBuild: C:\Windows\Microsoft.NET\Framework64\v4.0.30319\MSBuild.exe

Default parameters:

☐ Install automatically

Delete MSBuild

MSBuild Name: vs-msbuild

Path to MSBuild: C:\Program Files (x86)\Microsoft Visual Studio\2017\Enterprise\MSBuild\15.0\Bin\MSBuild.exe

Default parameters:

☐ Install automatically

Delete MSBuild

Add MSBuild

List of MSBuild installations on this system

## Configure Email Notification

- Browse to the Jenkins homepage e.g. <http://d4w0335g.houston.hp.com:8080/> (8080 is the default port)
- Navigate to Manage Jenkins\Configure System
- In the E-mail Notification section, Enter:

**E-mail Notification**

SMTP server: smtp1.hp.com

Default User E-mail Suffix: @hp.com

System Admin E-mail Address: ning.liu4@hp.com

Hudson URL: http://d4w0335g.houston.hp.com:8080/

Advanced...

## 端到端持续集成演练





## Create a job to build ASP.NET MVC Project

The job in below screenshot will pull source code from GitHub every two minutes and do the source code compile using msbuild.

The image shows two screenshots of the Jenkins job configuration interface for a job named 'dxc-github-service-agent'.

**Top Screenshot: General and Source Code Management tabs**

- General tab:**
  - Project name: dxc-github-service-agent
  - Discard old builds: ☐
  - GitHub project: ☒
  - Project url: https://github.houston.entsvcs.net/ServiceAgent/ServiceAgentWithHistory/
  - Options: This project is parameterized, Disable this project, Execute concurrent builds if necessary.
- Source Code Management tab:**
  - Repository type: Git
  - Repository URL: https://github.houston.entsvcs.net/ServiceAgent/ServiceAgentWithHistory.git
  - Credentials: ning.liu4@hpe.com/\*\*\*\*\*
  - Branches to build: \*/master

**Bottom Screenshot: Build Triggers tab**

- Build Triggers tab:**
  - Options: Trigger builds remotely (e.g., from scripts), Build after other projects are built, Build periodically, GitHub hook trigger for GITScm polling.
  - Poll SCM: ☒
  - Schedule: H/2 \* \* \* \*

In the configuration page of the job, Build->add build step -> Execute windows batch command...

Below is a sample example for our project.

```
@echo "Run Unit Test....."

@set
MySourceCodeWorkSpace=D:\\Program\\Jenkins\\workspace\\dxc-github-
service-agent

@set
MyUnitTestJobWorkSpace=D:\\Program\\Jenkins\\workspace\\dxc-
github-service-agent-unit-test\\

if exist %MyUnitTestJobWorkSpace%\\TestResults == false

md %MyUnitTestJobWorkSpace%\\TestResults

rd /s /q %MyUnitTestJobWorkSpace%\\TestResults\\

md %MyUnitTestJobWorkSpace%\\TestResults

cd "D:\\Program\\Microsoft Visual
Studio\\2017\\Professional\\Common7\\IDE\\"

.\\MSTest.exe

/testcontainer:%MySourceCodeWorkSpace%\\SA.Tests\\bin\\Debug\\SA.T
ests.dll
/testcontainer:%MySourceCodeWorkSpace%\\SA.PdM.BL.Tests\\bin\\Debu
g\\SA.PdM.BL.Tests.dll
/testcontainer:%MySourceCodeWorkSpace%\\CommonUtils.Tests\\bin\\De
bug\\CommonUtils.Tests.dll

/testcontainer:%MySourceCodeWorkSpace%\\SA.Service.Tests\\bin\\Deb
ug\\SA.Service.Tests.dll
```

Jenkins > dxc-github-service-agent-unit-test >

General Source Code Management Build Triggers Build Environment Build **Post-build Actions**

### Post-build Actions

**Publish MStest test result report**

Test report TRX file:

A path relative to [the workspace root](#), an Ant fileset pattern, or an environment variable.

Fail build if no files are found: ☒

Retain long standard output/error: ☐

**Build other projects**

Projects to build:

☒ Trigger only if build is stable  
☐ Trigger even if the build is unstable  
☐ Trigger even if the build fails

**E-mail Notification**

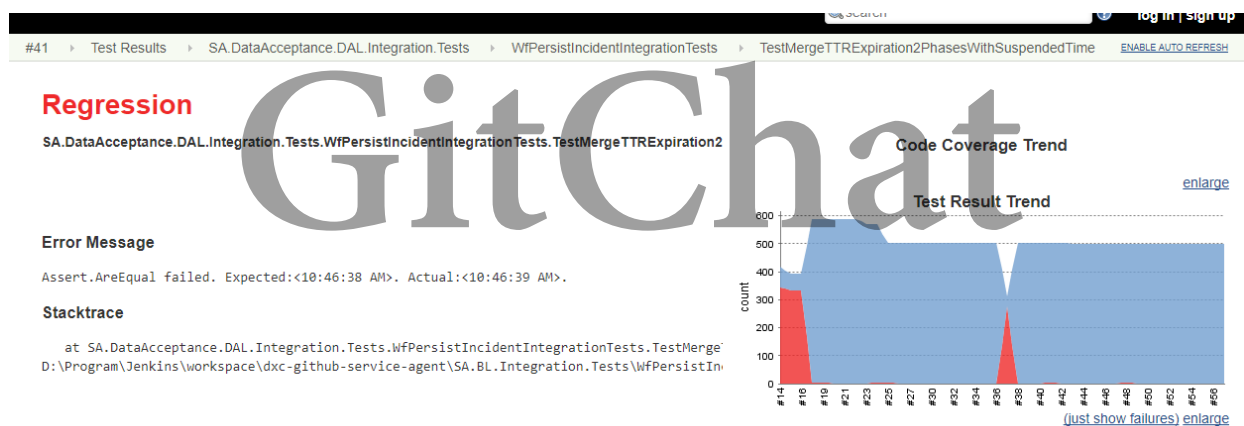
Recipients:

Whitespace-separated list of recipient addresses. May reference build parameters like \$PARAM. E-mail will be sent when a build fails, becomes unstable or returns to stable.

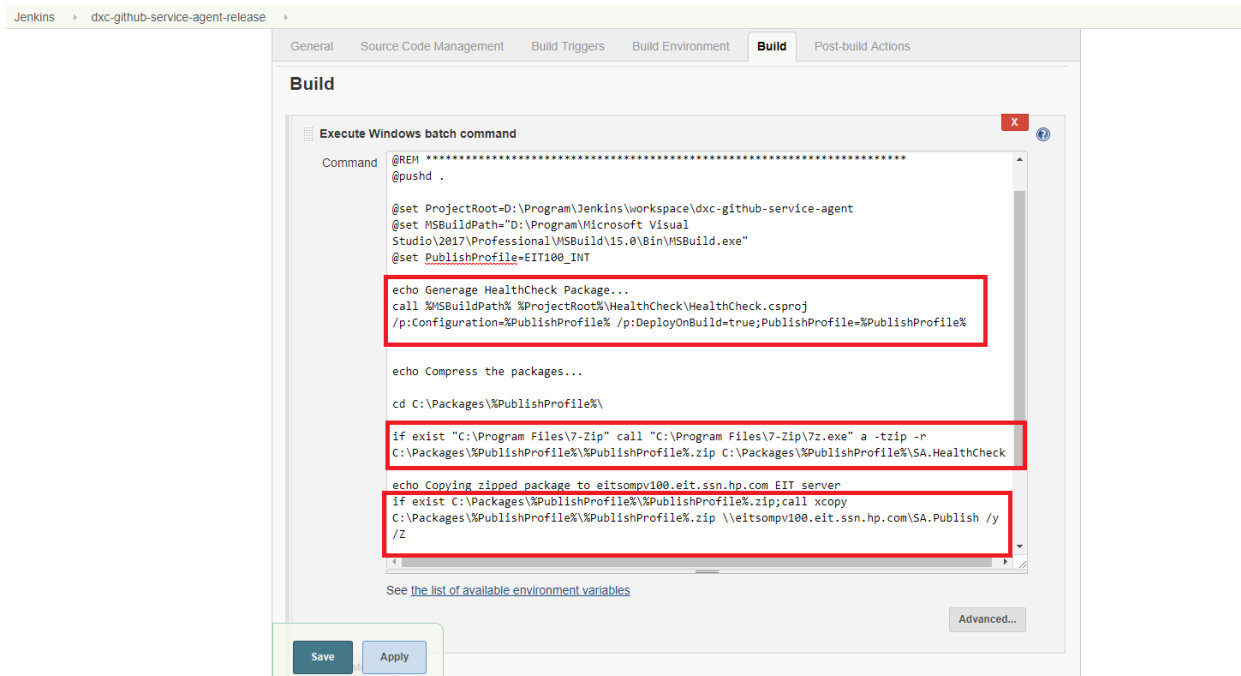
☒ Send e-mail for every unstable build  
☐ Send separate e-mails to individuals who broke the build

Save Apply

Below are sample result of the unit test job:



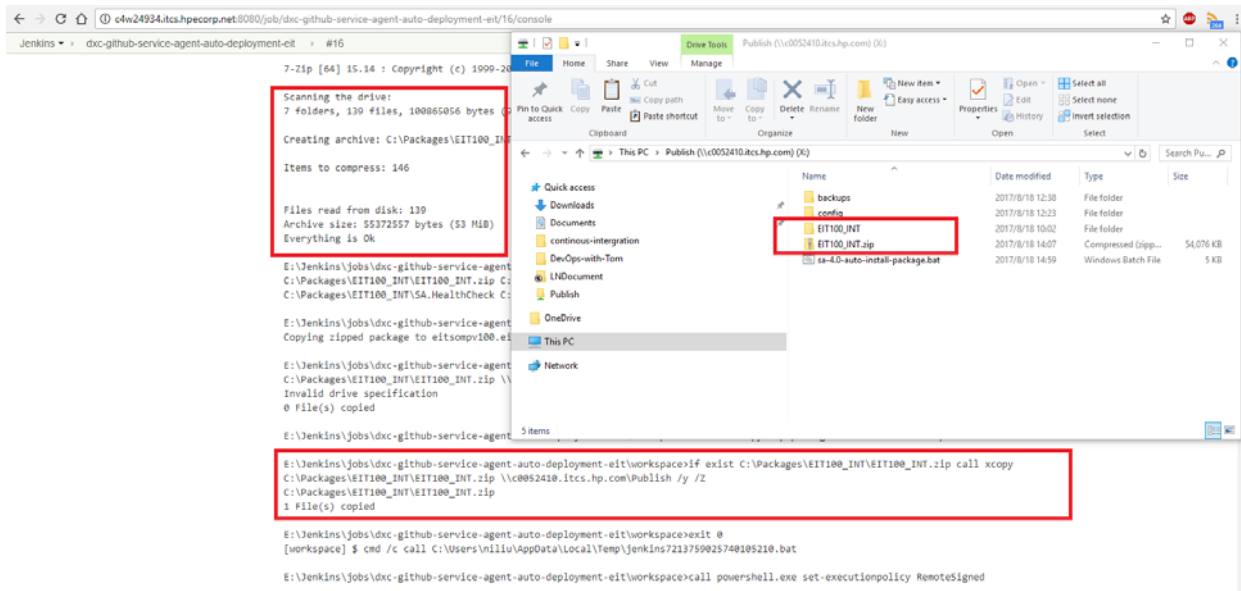
Create a job to generate and release package



For the batch script above, you can also manage it as git source code, maintained directly as code.

Then configuration will similar like below:





Create a job to install package and db script upgrade

DB scripts is managed as GitHub source code.

ServiceAgent-master > ServiceAgentWithHistory > DevOps > Database-scripts

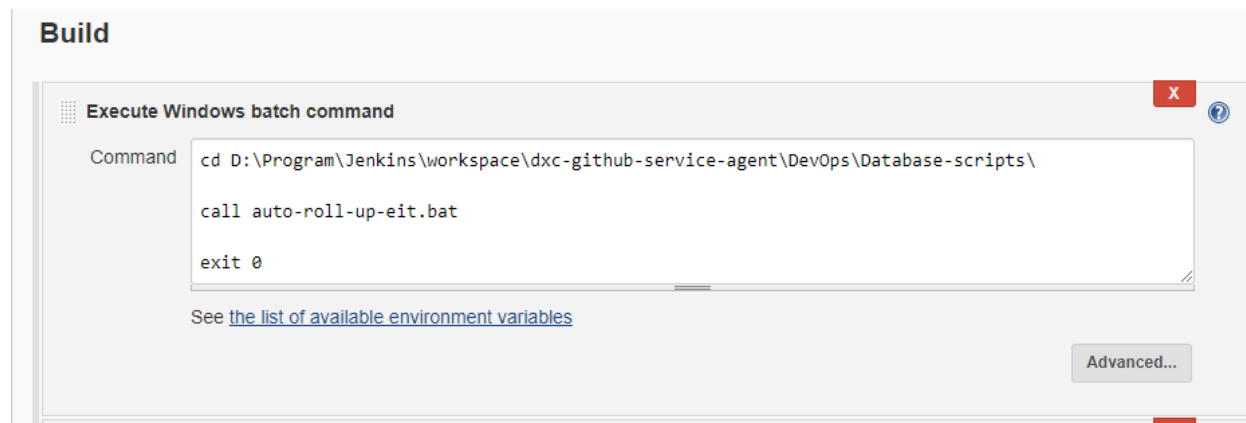
Name	Date modified	Type	Size
001-SA40-prod-patch13-ServiceAgent.sql	2017/9/13 12:55	Microsoft SQL Ser...	1 KB
002-SA40-prod-patch13-SADailyUsage_records.sql	2017/9/13 13:25	Microsoft SQL Ser...	16 KB
auto-roll-up-eit.bat	2017/9/13 13:25	Windows Batch File	1 KB

Every time, you add a new script file, update the auto-roll-up.bat.

To include the new added script file.

Then commit the new added script file and updated auto-roll-up.bat.

Below configuration of the installation job is in charge of the db roll upgrade.



Below configuration of the installation job is in charge of the remotely application installation.

Jenkins server run remote script from Jenkins servers using tool psexec.exe.

So now Jenkins can do continuously delivery to the app server when there is a code change.

(Two others ways I implemented to do remote installation, once is using powershell, another is through ansible, but will not cover in this doc)



Create a job to verify the app using specflow

Create a specflow test project to auto detect the healthy status of the application. Integrate the specflow project as a Jenkins job for verifying if the build is good enough.

For more info about specflow, please check [SpecFlow](#).

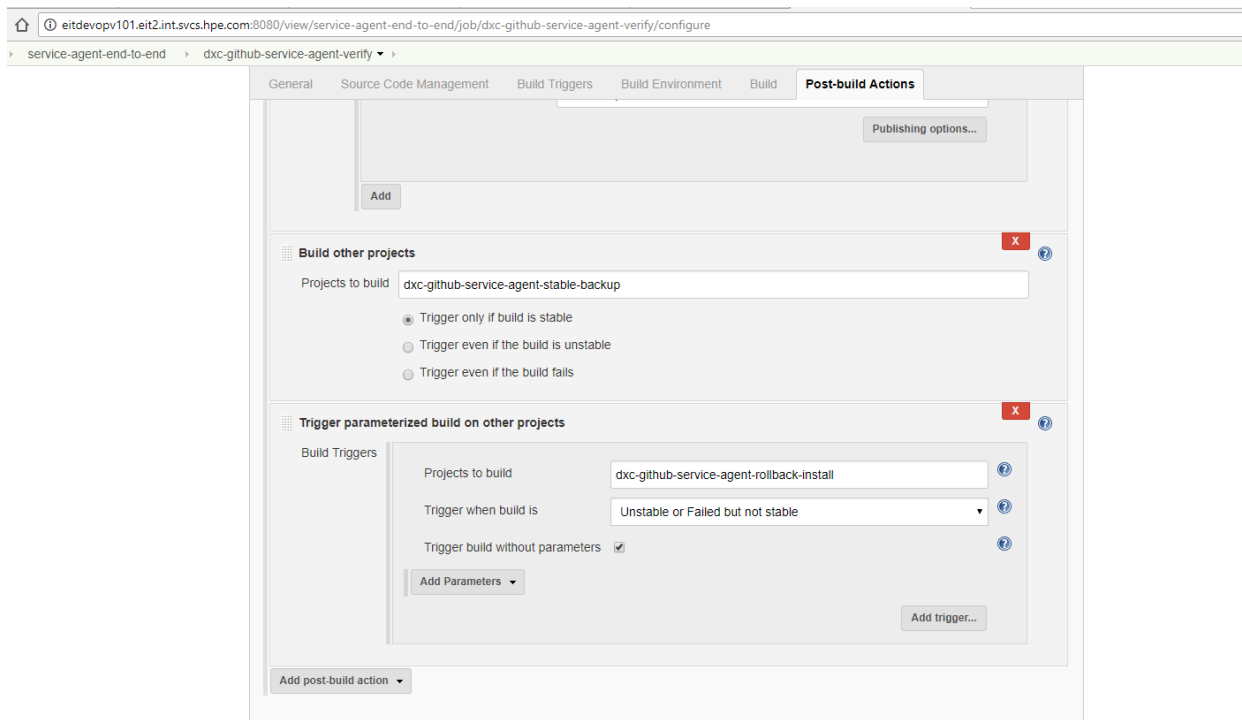
在specflow 项目里，你可以定义一些验证性测试场景用例，它会模拟人工去做smoke test 和acceptance test 来决定这次的部署是不是足够stable.

下面是我们项目中一个示例验证步骤场景：

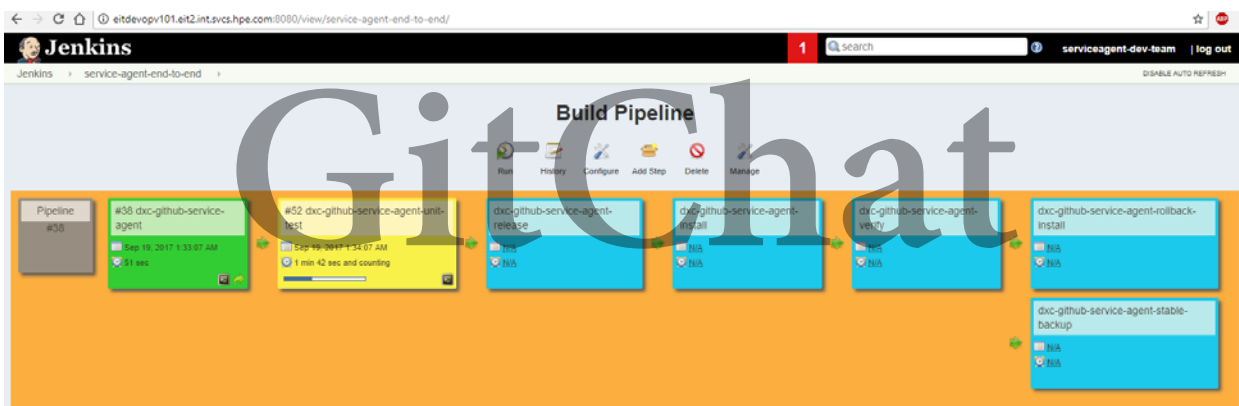
Create a job to auto backup and rollback

If the verification job of specflow test project pass, then back up the app packages as a latest stable version for future use.

If the [SpecFlow](#) verification job failed, then rollback the app using the last stable version.



## Final Pipeline



Demo the normal install and db roll upgrade

Check Points:



Demo the unstable install and roll back

Code change to simulate failure.

Change `.\healthcheck\controllers\homecontroller.cs` line 436 and commit to github.

```
if (ctx.SALoadStatus.Count() < 0)
```

to simulate a healthcheck status checkpoint failure.

Check Points:

1. rollback script applied

```
select \* from [dbo].[SAApplcationSettings] where Name =  
'MassUpdateTicketsInParallel'
```

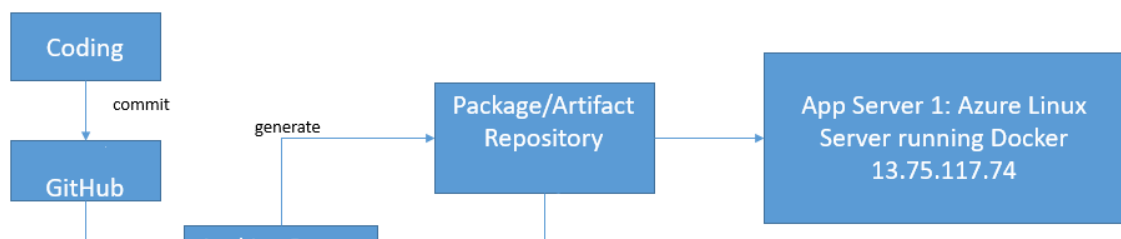
2. Before rollback start, health check page eit100 with red

3. After rollback finished, health check page eit100 rollback to green

自动部署with-docker-ansible

# GitChat

Below is a flowchart of deployment with docker and ansible



# 演示录像

链接: <https://pan.baidu.com/s/1i5eQshn> 密码: 2upi

## 一些项目实践经验

### Best Practices

Normal for single project, we create 4 Jenkins jobs to help the development.

1. Quick Break Build Check— every 5mins, if change made into SVN, then auto trigger the build.
2. Daily Job– two times every day 8:00 and 17:00, get SCM code, build code, unit testing, code quality analyze.
3. Full job – one time every Friday, get SVN code, build code, unit test, code quality analyze, source code API documentation, deployment.
4. Temporary job – target branches code for investigation purpose .(optional)

经验教训 lesson learn

Windows server 2008 R2 上面不能安装visual studio 2015 或更新版本，所以如果你想在Jenkins server 上直接安装visual studio, 至少Jenkins server版本要Window server 2012

### Troubleshooting

```
sourceanalyzer -b SA_3.2
"C:\\Windows\\Microsoft.NET\\Framework\\v4.0.30319\\MSBuild.exe"
E:\\Jenkins\\jobs\\SATrunkBreakBuild\\workspace\\SA_3.2.sln
/t:rebuild
/p:Configuration=Debug;TargetFrameworkVersion=v4.5
```

Execute Windows batch command

```
命令
@REM Generate Report in html format.
E:
cd "E:\Program Files\HP_Fortify\HP_Fortify_SCA_and_Apps_4.00\bin"
sourceanalyzer -b SA_3.2 -clean
sourceanalyzer -b SA_3.2 "C:\Windows\Microsoft.NET\Framework\v4.0.30319\MSBuild.exe" E:\Jenkins\jobs\SATrunkBreakBuild\workspace\SA_3.2.sln /t:rebuild
/p:Configuration=Debug;TargetFrameworkVersion=v4.5
sourceanalyzer -b SA_3.2 -scan -f E:\Jenkins\jobs\SATrunkBreakBuild\workspace\FortifyReport\index.fpr -format fpr -html-report
```

[参阅 可用环境变量列表](#)

MSBuild Error : MSB4175: The task factory "CodeTaskFactory" could not be loaded from the assembly

Details error: MSB4175: The task factory "CodeTaskFactory" could not be loaded from the assembly "C:\\Program Files (x86)\\MSBuild\\12.0\\bin\\Microsoft.Build.Tasks.v4.0.dll". Could not load file or assembly 'file:///C:\\Program Files (x86)\\MSBuild\\12.0\\bin\\Microsoft.Build.Tasks.v4.0.dll' or one of its dependencies. The system cannot find the file specified.

Root Cause :

Somebody when create the project, the tools version target to 12.0, which means using the MSBuild

Along with visual studio 2013 tool which located at C:\Program Files(x86)\MSBuild\12.0\bin

MSBuild

Name: msbuildwithdotnetframework

Path to MSBuild: C:\Windows\Microsoft.NET\Framework64\v4.0.30319\MSBuild.exe

Default parameters:

☐ Install automatically

[Delete MSBuild](#)

[Add MSBuild](#)

List of MSBuild installations on this system

## Solution:

1. If in the Jenkins server, you have the visual studio 2013 installed, or you have standalone Microsoft Build Tools 2013 from <https://www.microsoft.com/en-us/download/details.aspx?id=40760> installed, you can update the msbuild path

MSBuild

Name: MSBuild

Path to MSBuild: C:\Program Files (x86)\MSBuild\14.0\Bin\MSBuild.exe

Default parameters:

☐ Install automatically

[Delete MSBuild](#)

2. If you want to depend on dotnet framework only, you can update \*.csproj file which report the error to set the ToolsVersion from 12.0 to 4.0

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <Project ToolsVersion="12.0" DefaultTargets="Build" xmlns="http://schemas.microsoft.com/developer/msbuild/2003">
3   <PropertyGroup>
4     <Configuration Condition="$([Configuration]) == 'Debug'">Configuration</Configuration>
5     <Platform Condition="$([Platform]) == 'AnyCPU'">Platform</Platform>
6     <ProjectGuid>{A09DCF77-5E58-45CA-9B8E-6113081F66B7}</ProjectGuid>
7     <OutputType>Library</OutputType>
8     <AppDesignerFolder>Properties</AppDesignerFolder>
9     <RootNamespace>TimeTrackingServiceUnitTest</RootNamespace>
10    <AssemblyName>TimeTrackingServiceUnitTest</AssemblyName>
11    <TargetFrameworkVersion>v4.5</TargetFrameworkVersion>
12    <FileAlignment>512</FileAlignment>
13    <ProjectTypeGuids>{3AC096D0-A1C2-E12C-1390-A8335801FDAB};{FAE04EC0-301F-11D3-BF4B-C08F59BB2646}</ProjectTypeGuids>
14    <VisualStudioVersion Condition="$([VisualStudioVersion]) == '10.0'">10.0</VisualStudioVersion>
15    <VSToolsPath Condition="$([VSToolsPath]) == 'MSBuildExtensionsPath32'">$(MSBuildExtensionsPath32)\Microsoft\VisualStudio\10.0</VSToolsPath>
16    <ReferencePath>$(ProgramFiles)\Common Files\microsoft shared\VSTT\$(VisualStudioVersion)\UITestTools</ReferencePath>
17    <IsCodedUITest>False</IsCodedUITest>
18    <TestProjectType>UnitTest</TestProjectType>

```

12

NuGet can use local settings for its behavior which can be unpredictable if you're not 100% sure how the server is configured.

I prefer putting the NuGet settings inside the `<sln root>/nuget/NuGet.targets` file which is version controlled and at a single location. I got this working with 3 quick edits to `<sln root>/nuget/NuGet.targets`, they should look as below after editing:

#### Change 1:

```
<!-- Enable the restore command to run before builds -->
<RestorePackages Condition=" '$(RestorePackages)' == '' ">true</RestorePackages>
```

#### Change 2:

```
<!-- Determines if package restore consent is required to restore packages -->
<RequireRestoreConsent Condition=" '$(RequireRestoreConsent)' != 'false' ">>false</RequireRestoreConsent>
```

My comment: Awkward logic but think of "requires consent not equal to false must be true" (original) as "requires consent equal to true must be true" (translated) and it makes sense to change the last part to "false" (the edit)

**Change 3 :** I also added/uncommented the `<PackageSource ... >` tag to to remove any dependencies on the

```
<ItemGroup Condition=" '$(PackageSources)' == '' ">
  <PackageSource Include="https://nuget.org/api/v2/" />
</ItemGroup>
```

<https://stackoverflow.com/questions/12788521/nuget-package-restore-not-working-on-build-server>

# GitChat

Issue : Specflow report Error:

WatiN.Core.Exceptions.ElementNotFoundException: Could not find INPUT (hidden) or INPUT (password)

Root Cause: IIS failed to run with error below:

```
[SecurityException: Request for the permission of type
'System.Security.Permissions.SecurityPermission, mscorlib,
Version=4.0.0.0.
```

```
\<trust level="Full" /\>
```

```
\</system.web\>
```

Solution: Jenkins running a user don't have the db access.

Run below script in sql.

```
CREATE LOGIN [americas\ssadev001] FROM WINDOWS
```

Issue : Specflow report Error: WatiN.Core.Exceptions.RunScriptException:  
RunScript failed —> System.UnauthorizedAccessException

Solution: Adding the test site to the IE's trusted sites list and it has resolved the error message.

Issue : Specflow issue: the 'validation' attribute must be one of these values:  
SHA1

Parser Error Message: When **using** \<machineKey  
compatibilityMode="Framework45"  
/\> or the MachineKey.Protect and MachineKey.Unprotect APIs, the  
'validation'  
attribute must be **one** of these values: SHA1, HMACSHA256,  
HMACSHA384, HMACSHA512,  
or alg:[KeyedHashAlgorithm].

Solution:

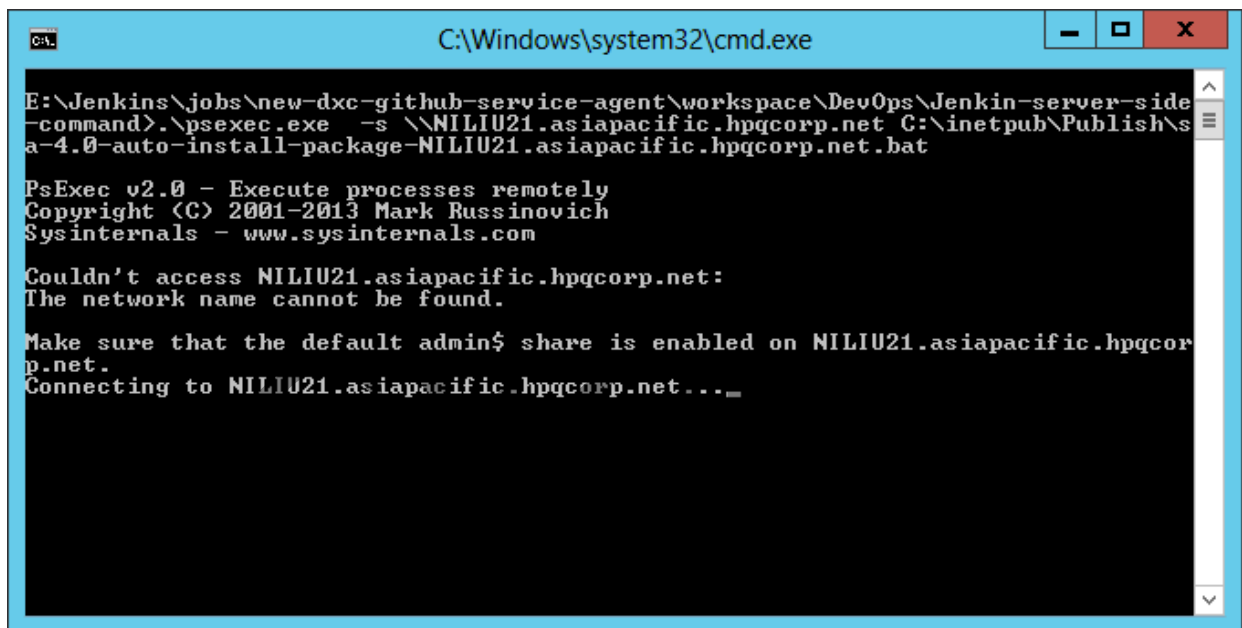
```
C:\\Windows\\Microsoft.NET\\Framework64\\v4.0.30319\\Config\\web.  
config line 400
```

Add the **-i** options highlighted below.

```
"E:\\Jenkins\\jobs\\new-dxc-github-service-agent\\workspace\\DevOps\\Jenkin-server-side-command\\psexec.exe"
-i -h -s \\\\c0052410.itcs.hp.com D:\\Publish\\sa-4.0-auto-install-package.bat
```

Issue : Psexec return error code 6 The handle is invalid.

Details: when run in the commandline, will see the details error, caused by admin\$ share not set,

A screenshot of a Windows command prompt window titled "C:\Windows\system32\cmd.exe". The window has a blue title bar and standard Windows window controls. The command prompt shows the following text:

```
E:\Jenkins\jobs\new-dxc-github-service-agent\workspace\DevOps\Jenkin-server-side-command>.psexec.exe -s \\NILIU21.asiapacific.hpqcorp.net C:\inetpub\Publish\sa-4.0-auto-install-package-NILIU21.asiapacific.hpqcorp.net.bat
```

Below the command, the output of PsExec is displayed:

```
PsExec v2.0 - Execute processes remotely
Copyright (C) 2001-2013 Mark Russinovich
Sysinternals - www.sysinternals.com

Couldn't access NILIU21.asiapacific.hpqcorp.net:
The network name cannot be found.

Make sure that the default admin$ share is enabled on NILIU21.asiapacific.hpqcorp.net.
Connecting to NILIU21.asiapacific.hpqcorp.net...
```

Follow:

<https://stackoverflow.com/questions/18388381/make-sure-that-the-default-admin-share-is-enable-on-servername>

Solution:

<http://aangelov.com/2014/08/30/getting-started-jenkins-msbuild-nuget-git/>

[www.martinfowler.com/articles/continuousIntegration.html](http://www.martinfowler.com/articles/continuousIntegration.html)

<http://redsolo.blogspot.com/2008/04/guide-to-building-net-projects-using.html>

<http://martinfowler.com/articles/continuousIntegration.html>

<http://jenkins-ci.org/>

<https://jenkins.io>

Jenkins distributed build master/slave node

# GitChat