



# **.NET Framework v1.1 Support Document**

Verification Utility  
List of Updated Files

For IHS Applications

IHS AccuMap® | IHS AccuLogs®

June, 2009



# .NET Framework v1.1 Support Documentation

## IHS Inc.

### **Summary**

This document contains 2 sections:

1. .NET policy creation and verification utility. Explains usage.
2. A list of new files installed for the Jan. 06 .NET release.

### **Audience**

This document is intended for IT staff supporting or installing .NET enabled IHS applications. Some of the procedures described require Administrator privileges on the local workstation and on the network server, if a server installation is required.

The .NET policy creation and verification utility has been designed for use by users of AccuMap. It will provide a means to easily add a .NET security code group and to verify that the security policy will work with the January 2006 release of AccuMap.



## **Section 1: .NET Policy Creation and Verification Utility.**

The .NET Policy Creation and Verification Utility has been developed by IHS to aid in creating the required IHS Code Group, setting the appropriate security level and verifying that the security settings are correctly set for your environment.

The .NET utility is actually 3 separate executables:

*AddCodeGroup.exe* – An interactive utility that will add the IHS Code Group.

*SecurityTest.exe* – Checks for the .NET v1.1 Framework and launches a .NET assembly from the same directory.

*VerifyPolicy.exe* – A .NET assembly that tests the FullTrust permission of the IHS Code Group.

### AddCodeGroup.exe

*AddCodeGroup.exe* is a console application that adds the FullTrust permission for a named code group for a particular URL. The name of the code group is IHS Energy.

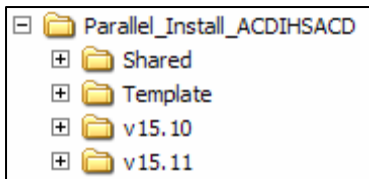
The URL defaults to the directory of the AccuMap Database Install:  
*[HKEY\_CURRENT\_USER\Software\AccuMap EnerData\RootDir]*.

This is where '*accu32.exe*' is located for a standard AccuMap install, or where '*accuwin.exe*' is located for a parallel AccuMap install. This URL can be overridden.

A Code Group description is also inserted and indicates that the Code Group was added by this utility. This utility should only be run once per machine or user combination otherwise identical copies of the Code Group will be created.



Below is a sample run where the parallel installation is on drive letter is P:



```
C:\>addcodegroup

addcodegroup      ($Rev: 1509 $)

The location of the AccuMap parallel installation is:
P:\Parallel_Install_ACDIHSACD\
Yes/No (default will be Yes, "quit" to exit program): yes

Microsoft (R) .NET Framework CasPol 1.1.4322.573
Copyright (C) Microsoft Corporation 1998-2002. All rights
reserved.

Added union code group with "-url" membership condition to the
Machine level.
Success

The following code group has been added successfully:
IHS Energy -- P:\Parallel_Install_ACDIHSACD\

Press any key to exit...
```



### SecurityTest.exe

*SecurityTest.exe* is a console application that tests to see if you have the proper .NET framework installed, and if so, if it would be able to launch a .NET assembly (the future *accu32.exe*) if it were located in the same directory as this application. It returns a code of zero if everything went okay and non-zero if there was a problem. The application has a few command line options to make it more flexible.

*/q* – Will run in quiet mode. Without this the application will display a popup letting the user know everything is okay or what they will need to address to allow the new .NET AccuMap to run. The return code is unaffected by this option (IE: The application will still exit with the same return code, regardless of whether */q* is specified or not).

*/l=[logfile]* – Will append the resulting success or failure message to the specified logfile. Without this, you will still get the popup dialog if you haven't used the */q* switch, this switch is simply a method to record the information. The logfile will be created if it's not already there. If the application cannot write to the logfile, you will be notified of this and no message will be recorded. This can be used in conjunction with */q* to capture the result of the test without having to interactively dismiss the popup message.

*/d=[folder]* – Record the logfile information to a logfile, but directs the file to be created in the specified directory \*and\* name the files using the following convention:

*[SuccessIndicator]\_[MachineName]\_[UserName].log.*

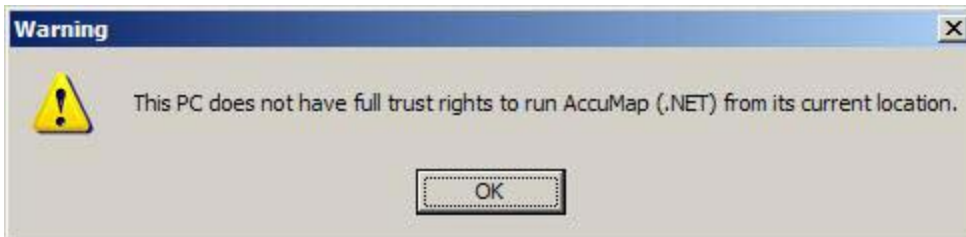
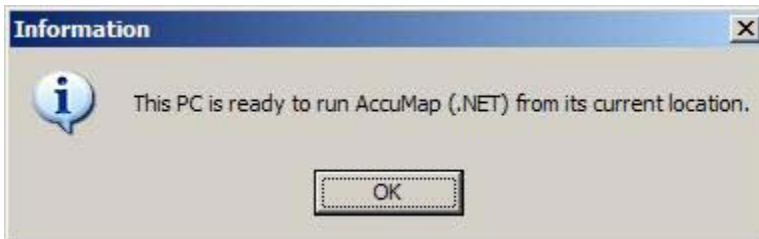
Where "SuccessIndicator" is "Y" or "N", and the Machine Name and User Name are derived from the Windows environment.

For example, user 'HSimpson' logged on to computer 'Reactor1' successfully passed the security test will result in a logfile named; Y\_Reactor1\_Hsimpson.log in the folder specified.

*/?* – Displays the usage information.



Below are the dialogs that will be displayed in interactive mode:



### Examples

*SecurityTest /q /d=<server>\temp*

*SecurityTest /q /l=C:\temp\verify.log*

(Paths that contain spaces must be enclosed in "quotations")



### VerifyPolicy.exe

A console application that returns a zero if it has got the FullTrust rights to run the AccuMap .NET assembly. It will return a 1 if it was unsuccessful.

*VerifyPolicy.exe* is called by *SecurityTest.exe* during its execution. *VerifyPolicy.exe* must reside in the same location as the *accuwin32.exe* executable otherwise it will not be verifying the permissions that *accu32* requires.

### **Location of Utilities and Run Procedure**

To run any of the utilities, you can either double-click on them from Windows Explorer or run them from a Command Prompt.

1. Copy the 3 executables to the location of *accu32.exe*.
2. From the workstation, run *AddCodeGroup* to add the IHS code group
3. From the workstation, run *SecurityTest* to verify the FullTrust permission.

### **Rollout Suggestions:**

#### **Silent Command Line Install of the .NET v1.1 Framework**

```
dotnetfx.exe /q:a /c:"install /l /q"
```

Specifying the */q:a* and */q* options for a silent installation allows for a standardized user installation. Specifying the */l* option creates a setup log file, *Netfx.log*, in the *%temp%* directory where all errors are logged.

#### **Automate Security Settings**

Add the following line to a login script to automate the security configuration. Edit the *file://* path to reflect your installation. This script will not check for the presence of an existing code group and if run multiple time, will create multiple entries. Ensure it only runs once.

```
%WINDIR%\Microsoft.NET\Framework\v1.1.4322\caspol -quiet -machine -  
addgroupAll_Code -url file://T:/IHSEnergy/* FullTrust -n "IHS Energy"
```

#### **Verify Workstation Readiness during Login**

Add the command below to your login script

```
SecurityTest /q /d=<server>\DotNetLogs
```



Sample Batch file to verify security policy and add it if it is not set

T:\VHSEnergy is for sample purposes only. Change the path to reflect your actual environment.

**Note:** It is assumed that .NET v1.1 is already installed.

This will run the VerifyPolicy executable to determine if the IHS Energy Code Group has been installed. If not, it will run the CASPOL utility and add the Code Group. Administrator rights are required to add the Code Group.

```
REM =====
REM Test and set security settings for IHS Energy
REM =====

REM Run the test in quiet mode
T:\IHSEnergy\verifypolicy /q

REM Optionally log results to the folder T:\IHSEnergy\DotNetResults
REM T:\IHSEnergy\verifypolicy /q /d= T:\IHSEnergy\DotNetResults

REM If return code =1 (Failure), create Code Group
if errorlevel 1 goto setallcode
goto finish

:setallcode
%WINDIR%\Microsoft.NET\Framework\v1.1.4322\caspol -quiet -machine -
addgroup
All_Code -url file:///J:/accumap/* FullTrust -n "IHS Energy"

:finish
```

Using CASPOL to add the IHS Energy Code Group

To prevent duplicate IHS Energy code groups from being created on multiple run, reset all Machine Policies before adding the IHS policy.

```
%WINDIR%\Microsoft.NET\Framework\v1.1.4322\caspol -pp off -reset
```

Adding the IHS Energy Code Group

```
%WINDIR%\Microsoft.NET\Framework\v1.1.4322\caspol -quiet -machine -
addgroupAll_Code -url file:///T:/IHSEnergy/* FullTrust -n "IHS Energy"
```

To use UNC path use file:///<ServerName>/<Sharename>



## Silent Command Line Install of the .NET v1.1 Framework

```
dotnetfx.exe /q:a /c:"install /l /q"
```

Specifying the /q:a and /q options for a silent installation allows for a standardized user installation. Specifying the /l option creates a setup log file, Netfx.log, in the %temp% directory where all errors are logged.

## Using SecurityTest '/l' Switch to Capture More Information

```
echo %TIME% %DATE% Computer:%COMPUTERNAME% User: %USERNAME% >>  
%TEMP%\log.txt &^  
SecurityTest.exe /q /l=%TEMP%\log.txt &^  
echo %ERRORLEVEL% >> %TEMP%\log.txt
```

## Using Group Policies to distribute and configure.NET 1.1

First you will need to download the Microsoft.NET v1.1 redistributable from Microsoft's website. Next, Follow the instructions on this website to push out .NET to each workstation in the domain:

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnnetdep/html/deploynetfx1-1withad.asp>

Now run the following command on a workstation that will need AccuMap (You will only need to run this once):

```
%WINDIR%\Microsoft.NET\Framework\v1.1.4322\caspol -quiet -machine -addgroup  
All_Code -url file://P:/Accumap/* FullTrust -n "IHS Energy"
```

You will need to change the url to point to the location of your network AccuMap installation.

Now check to make sure that the change worked. Start up the .Net Configuration program from the Administrative Tools folder. Browse down the following tree: "Runtime Security Policy" > Machine > Code Groups > All\_Code > IHS Energy

If you see the IHS Energy entry you are all set, if you don't see it try running the caspol command again.

Now follow the steps on this site to create a .MSI file with the .NET settings for that client/site:

<http://support.microsoft.com/?scid=http%3a%2f%2fwww.support.microsoft.com%2fkb%2f815173%2fen-us%2f>



You can add this .MSI file to the same Group Policy you used to install .NET.

Using Group Policies will only install .NET and the AccuMap settings onto each PC once. Also if a user uninstalls .NET or removes the settings they will be automatically installed on the next reboot.

I've tested this and using these steps lets the AccuMap test programs run properly.

Also, the IHS support website talks some more about the .NET install and settings. You can find that site at:

<http://energy.ihs.com/Support/Documentation-CA/index.htm>

There are also some utilities that IHS has written that allow you to test the settings before the new AccuMap version comes out.



## Section 2: New File List

Below is a list of files that will be added to the same location as *accu32.exe*. These are .NET assemblies and are not shared by Windows.

DevExpress.Data3.dll  
DevExpress.Data3.xml  
DevExpress.Tutorials.dll  
DevExpress.Utils3.dll  
DevExpress.Utils3.xml  
DevExpress.XtraBars3.Design.dll  
DevExpress.XtraBars3.dll  
DevExpress.XtraBars3.xml  
DevExpress.XtraEditors3.Design.dll  
DevExpress.XtraEditors3.dll  
DevExpress.XtraEditors3.xml  
DevExpress.XtraGrid3.Design.dll  
DevExpress.XtraGrid3.dll  
DevExpress.XtraGrid3.xml  
DevExpress.XtraNavBar3.Design.dll  
DevExpress.XtraNavBar3.dll  
DevExpress.XtraNavBar3.xml  
DevExpress.XtraPivotGrid3.Design.dll  
DevExpress.XtraPivotGrid3.dll  
DevExpress.XtraPivotGrid3.xml  
DevExpress.XtraPrinting3.Design.dll  
DevExpress.XtraPrinting3.dll  
DevExpress.XtraPrinting3.xml  
DevExpress.XtraReports3.Design.dll  
DevExpress.XtraReports3.Import.dll  
DevExpress.XtraReports3.dll  
DevExpress.XtraReports3.xml  
DevExpress.XtraScheduler3.Design.dll  
DevExpress.XtraScheduler3.OutlookExchange.dll  
DevExpress.XtraScheduler3.VCalendarExchange.dll  
DevExpress.XtraScheduler3.dll  
DevExpress.XtraScheduler3.xml  
DevExpress.XtraTreeList3.Design.dll  
DevExpress.XtraTreeList3.dll  
DevExpress.XtraTreeList3.xml  
DevExpress.XtraVerticalGrid3.Design.dll  
DevExpress.XtraVerticalGrid3.dll  
DevExpress.XtraVerticalGrid3.xml  
log4net.dll  
IHSEnergy.Common.Persistence.dll  
IHSEnergy.Common.dll  
AccuMap.MapFeatures.dll



### **New Files Added to the Workstation**

*%APPDATA%\AccuMap\MapFeaturesControl\States.xml* where %APPDATA% is C:\Documents and Settings\%USERNAME%\Application Data

The above file is placed when the Map Features is launched. It is used to save the state of the dialog.