Intellicus DotNet Client Integration Guide

Intellicus Enterprise Reporting and BI Platform **Release 5.X**

Enterprise Reporting and Business Insights Platform

©Intellicus Technologies
info@intellicus.com
www.intellicus.com

Copyright © **2011** Intellicus Technologies.

This document and its content is copyrighted material of Intellicus Technologies. The content may not be copied or derived from, through any means, in parts or in whole, without a prior written permission from Intellicus Technologies. All other product names are believed to be registered trademarks of the respective companies.

Dated: - September 2011.

Acknowledgements

Intellicus acknowledges using of third-party libraries to extend support to the functionalities that they provide.

For details, visit: http://www.intellicus.com/acknowledgements.htm .

Table of Contents

Introduction

Intellicus functionalities are made available through Intellicus DotNet Client. It can be deployed within a host application.

A host application can call the Intellicus functionalities either by calling the Intellicus DotNet Client web pages (HTTP APIs) or by calling the provided methods (DotNet APIs).

These APIs interact with Intellicus Report Server, which actually carries out requested tasks.

We recommend reading chapter **Prerequisites** before actually going ahead with installation of Intellicus DotNet Client.

Important: Please read IntellicusGettingStarted.pdf before proceeding with installation of Intellicus DotNet Client.

Prerequisites

Intellicus DotNet Client has following prerequisites:

> **Important:** Intellicus DotNet Client is designed using .NET framework version 2.0.50727 SP1 (Microsoft Visual Studio 2005 Version 8.0.50727.42). It needs Internet Information Server (IIS) 5.1 or higher.

Internet Information Server (IIS)

.NET application needs Internet Information Server 5.1 or higher (IIS) to serve the web pages to clients. You need to make sure IIS is installed, is up and running before going ahead.

Browser

Intellicus DotNet Client is certified on MS Internet Explorer 8.0.

Host application

As stated in the earlier, Intellicus DotNet Client can be integrated with a host application. Make sure the application in which Intellicus client will be installed is already installed as a web site in IIS.

> **Note:** This document makes use of Sample Host Application to demonstrate steps of integration. If you are evaluating Intellicus DotNet Client, then install the SampleHostApplication before going ahead.

Intellicus Report Server

After integration, host application will access Intellicus functionalities. Make sure Intellicus Report Server is installed, up and running.

Installing Intellicus DotNet Client

Use this process to install Intellicus DotNet Client inside host application.

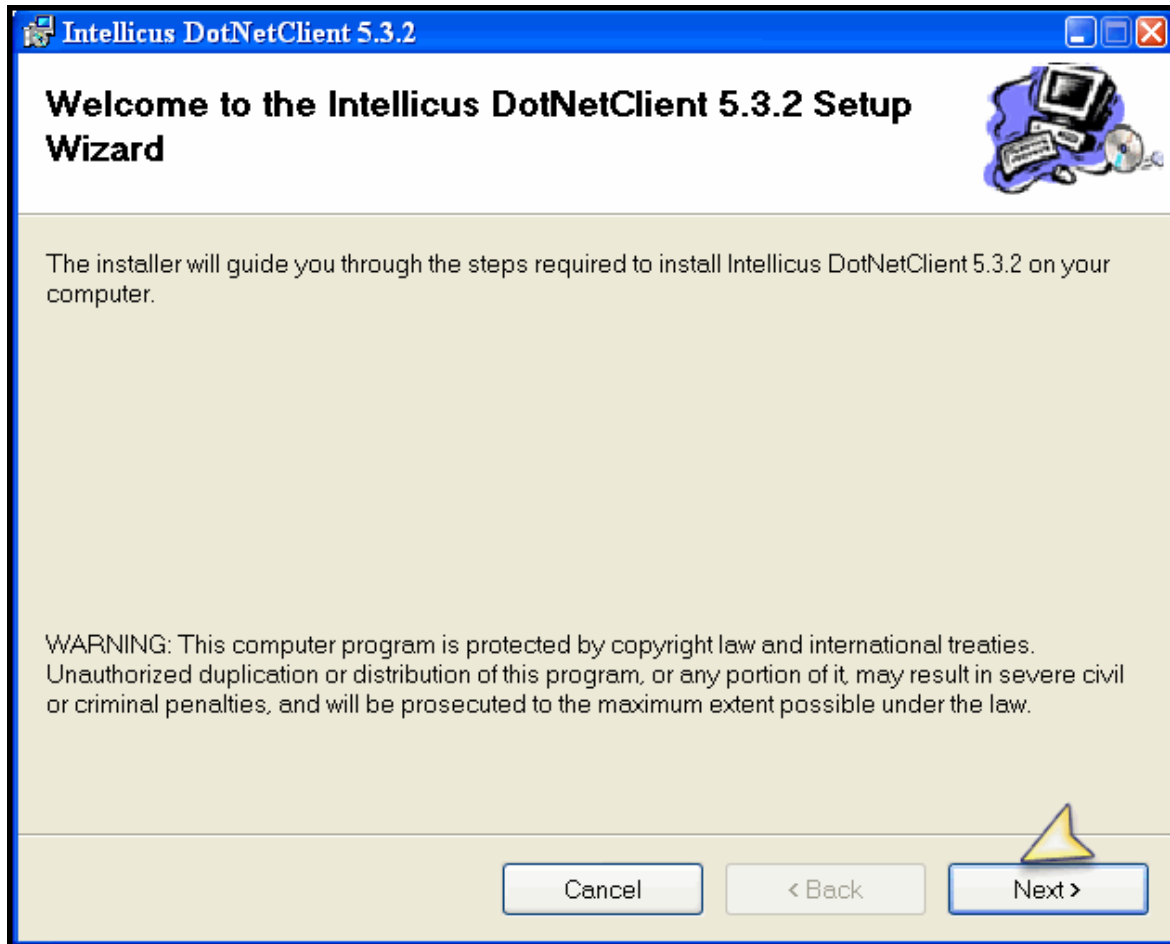**Step: 1. Execute Setup.exe file.**

Following window appears.

Figure 1: Welcome screen

Select Installation Folder step appears.

Step 2. Specify Destination Folder.

To deploy Intellicus as embedded in host application configured in IIS, destination path needs to be:

```
C:\Inetpub\wwwroot\<Host Application>\intellicus
```

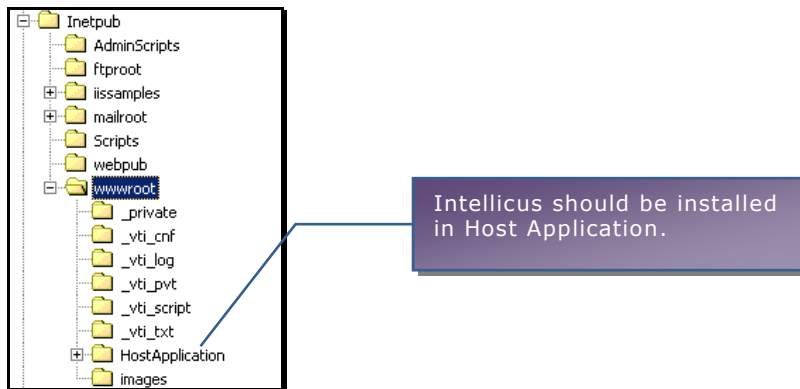<Host Application> in above path indicates application folder of host application.

Intellicus should be installed in Host Application.

Figure 2: Install Intellicus in Host Application

Intellicus needs to be installed in folder with name **intellicus** and in folder of the host application.

Using **Browse** button, go to the right path.
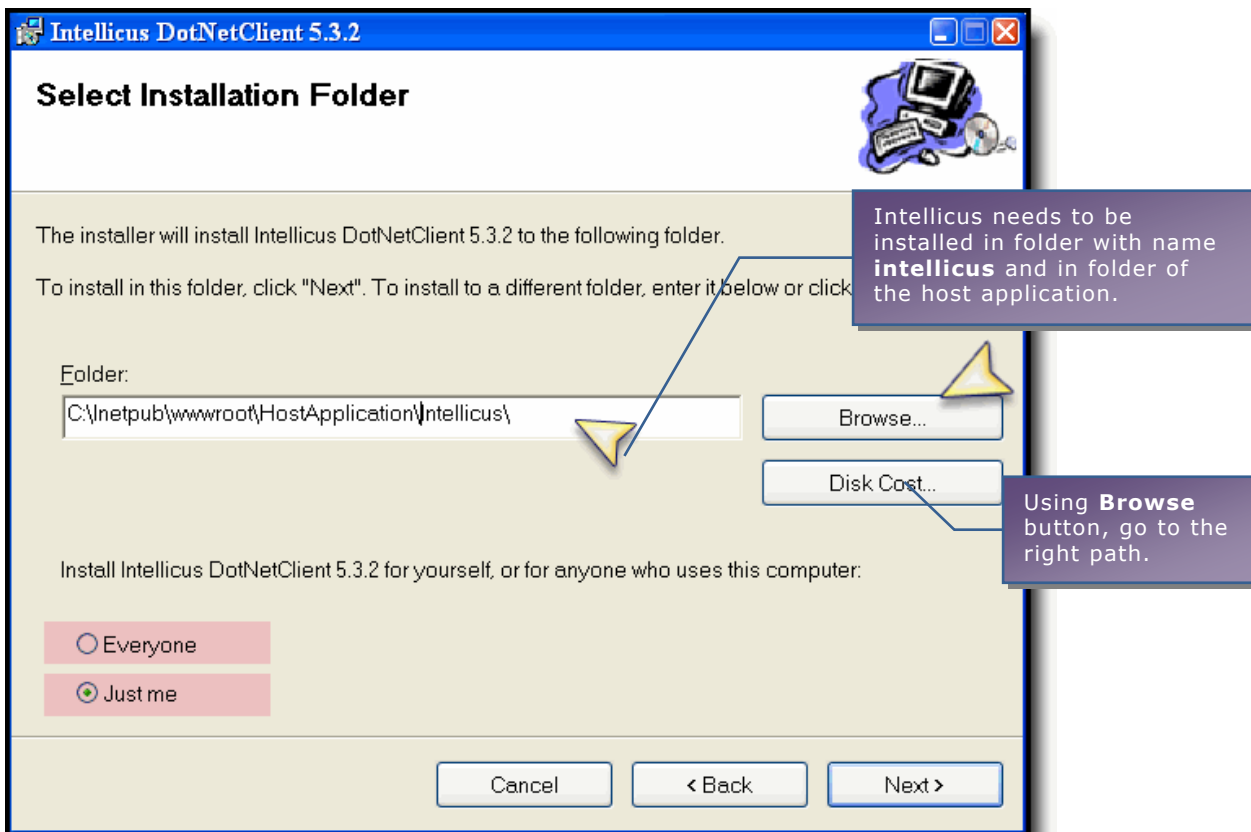
Figure 3:Select the Installation Folder

> **Important:** It is recommended that Intellicus application should be installed in folder "intellicus" and that folder is just inside folder of host application.

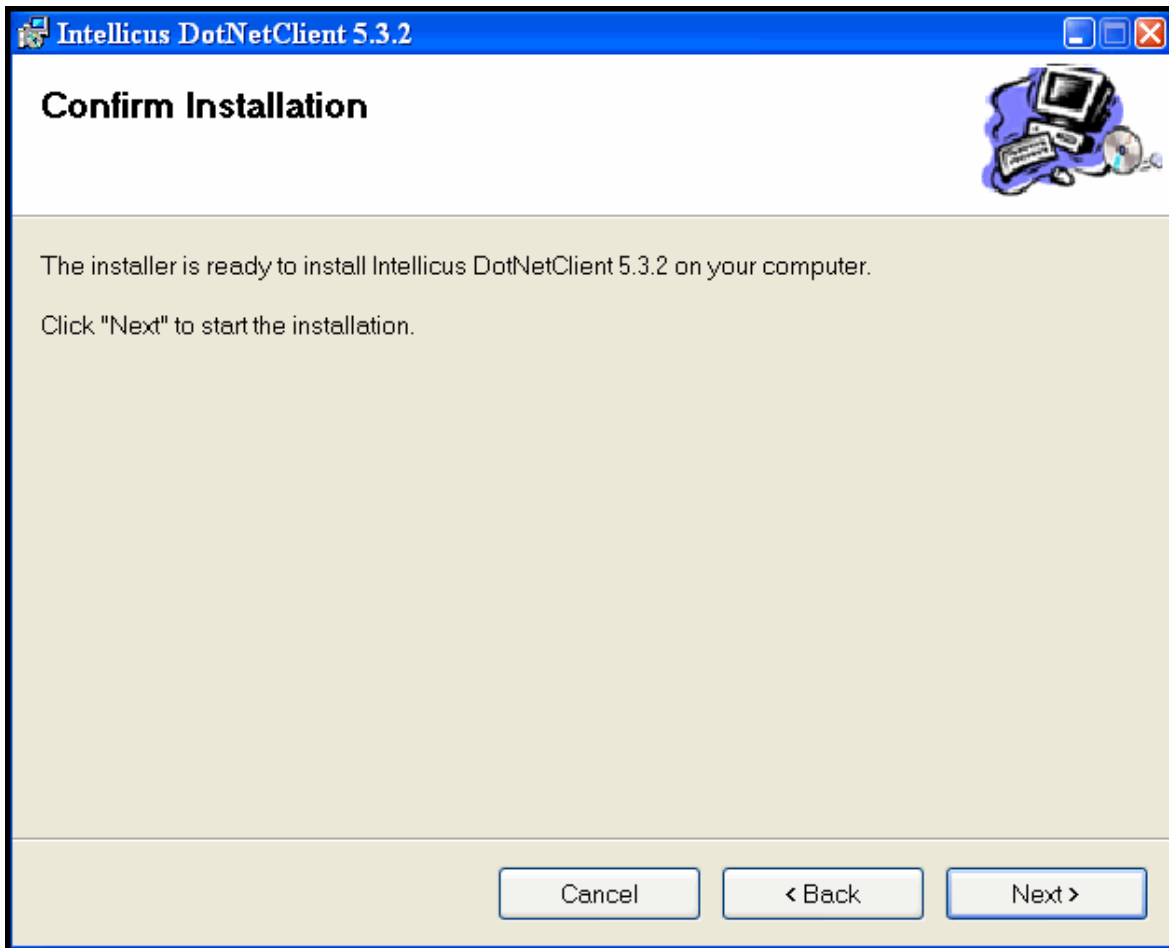Click **Next**. **Confirm Installation** screen appears.

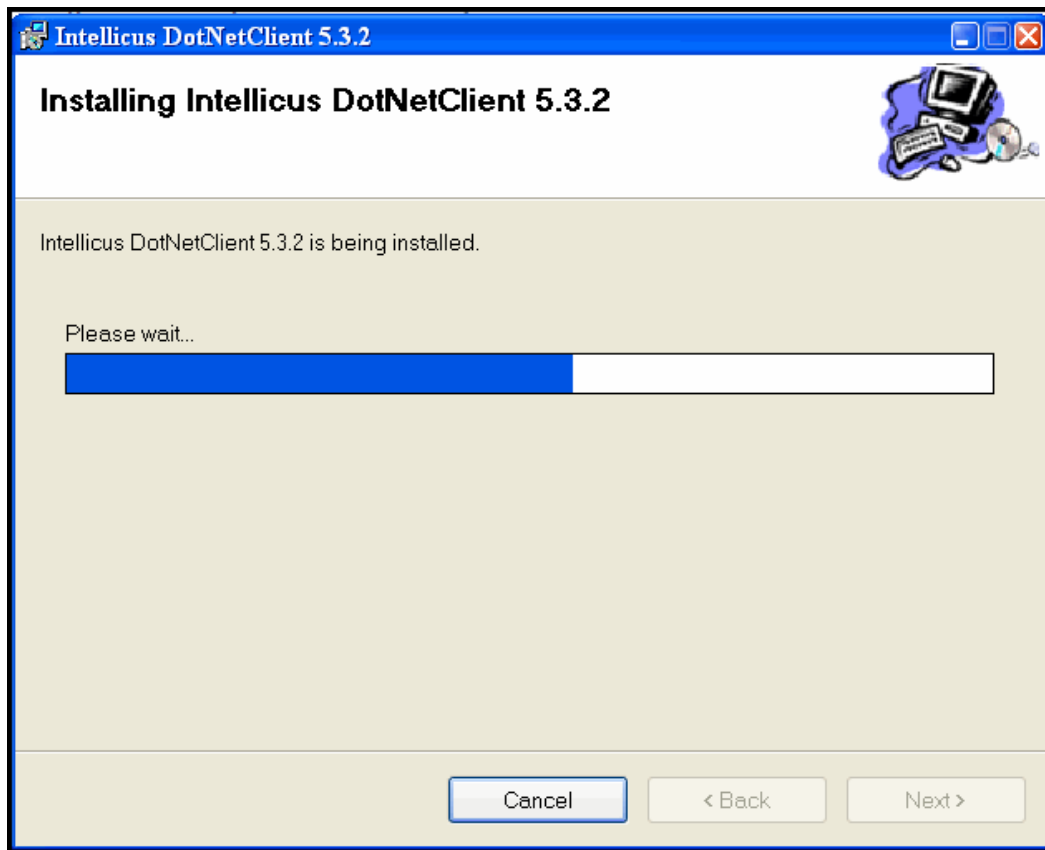Figure 4: Confirm Installation

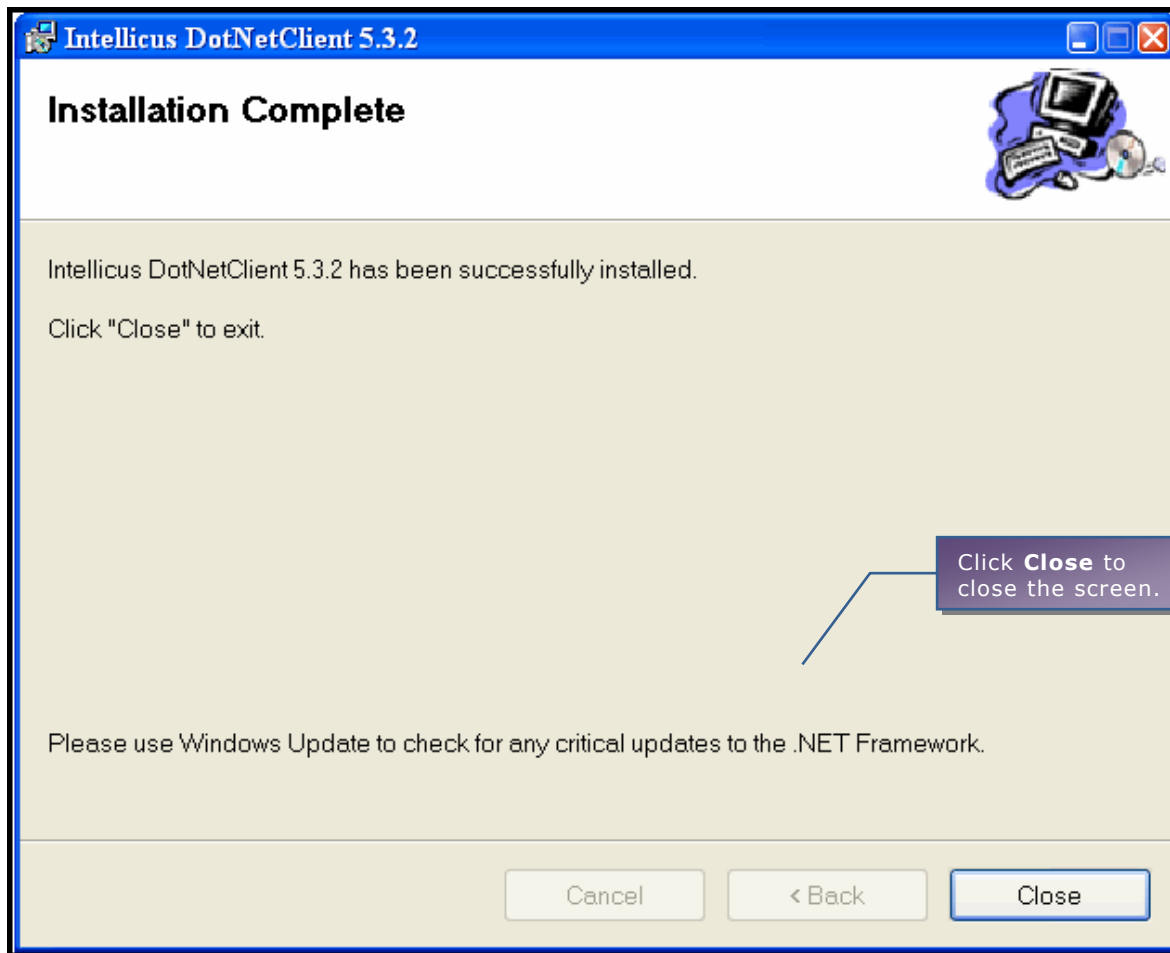Installation starts.

Figure 5:Installation is in process

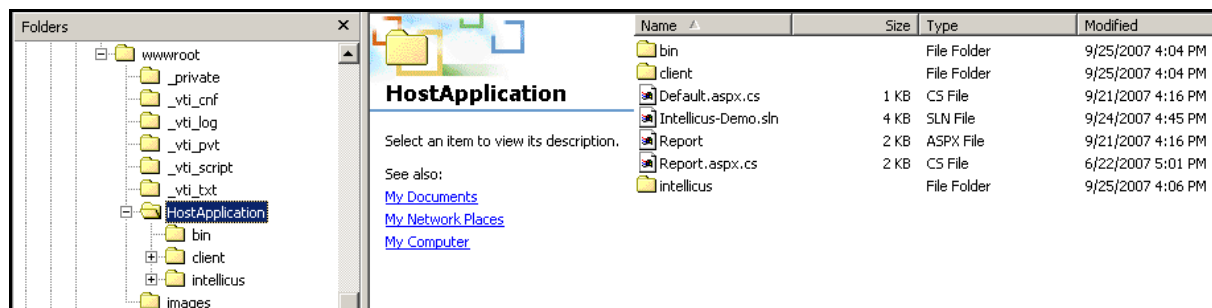Figure 6: Installation is complete

Figure 7: Intellicus Folder in Host Application Folder

Intellicus DotNet Client is installed.

Integrating Intellicus

After installing the Intellicus DotNet Client, next steps is to carryout activities that will make it possible to access Intellicus from within the host application.

The activities are:

- **Moving the binaries:** Moving dll files of Intellicus to bin folder of HostApplication.

- **Merging contents of global.asax file:** Merging contents of global.asax shipped with Intellicus to that of HostApplication.

- **Merging contents of Web.Config file:** Merging contents of Web.config shipped with Intellicus to that of HostApplication.

- **Handling Session timeout:** Specifying the URL of desired login page in case of session timeout.

- **Specifying Intellicus logs related information:** Specifying the logging details for Intellicus in Log4NetConfig.xml file.

- **Providing Access Privileges:** Providing access permission to worker process on application folders to read/write the data.

- **Setting IIS properties:** Adding extension mappings for host application and enabling anonymous authentication mode for Intellicus.

Moving dll files to bin of HostApplication

The /HostApplication/Intellicus/bin folder contains the dll files that Intellicus DotNet Client needs. These files need to be moved from their existing location to /HostApplication/bin.
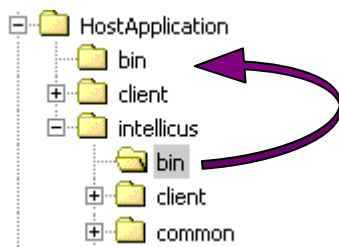

Figure 8:Bin Folder of Host Application

> **Note:** Delete /HostApplication/intellicus/**bin** folder after files are copied.

Merging contents of global.asax file

Following application initialization code should be placed within different events exist in global.asax file.

(Adding Intellicus.URLRewriter namespace in Global.asax file)

```
<%@ Import Namespace="Intellicus.URLRewriter"%>
```

(Adding selected contents of Global.asax file)

Add the content of following sections of /HostApplication/intellicus/Global.asax file to the content of /HostApplication/Global.asax file.

Application_Start

```
void Application_Start(object sender, EventArgs e)
    {
  //Relative path of Report Client properties file.
    string strPath = "/client/Config/ReportClient.properties";
        try
        {
          //Initialize the Intellicus application pre-requisites.
          Initial.InitializeApplication(Application, strPath);
        }
        catch (Exception ex)
        {
          //Handle the exception occured.
        }
```

Application_End

```
void Application End(object sender, EventArgs e)
    {
        //Perform Intellicus application cleanup task.
        Initial.CleanUpTask("");
    }
```

Application_BeginRequest

```
public void Application_BeginRequest(Object source, EventArgs e)
    {
```

```
      //Perform URL rewriting required for Intellicus application.
     string strIntellicusPath = Request.ApplicationPath;
     if (Request.Path.ToUpper().Contains
                             (strIntellicusPath.ToUpper()))
      {
             Rewriter.Process();
      }
    }
```

Session_End

```
void Session_End(object sender, EventArgs e)
    {
        //Perform Intellicus application cleanup task.
        Initial.CleanUpTask(Session.SessionID);
    }
```

Application_Error

```
void Session_End(object sender, EventArgs e)
{
    //Show  user  friendly  error  message  page  by  supressing  the
system error page.
        Server.ClearError();
    }
```

> **Important:** Delete /HostApplication/intellicus/Global.asax after this
> activity is completed.

Merging contents of Web.Config file

(Add contents of web.config file)

Application configuration information is stored in web.config file. We need to copy this information from **/HostApplication/Intellicus/**web.config to **/HostApplication/**web.config file.

Make sure contents enclosed within following tags are added:

sectionGroup

This tag is found inside <configSections> tag. You need to append information between sectionGroup tag start and end.

```
<configSections><!-Add this tag if not present in host app
web.config-->
      <sectionGroup name="system.web">
          <section name="IntellicusUrlRewriter"
              type="Intellicus.URLRewriter.Rewriter,
              Intellicus.URLRewriter"/>
      </sectionGroup>
</configSections>
```

If <sectionGroup> tag is not found in **/HostApplication/**Web.Config file, copy the content including start-tag and end-tag from **/HostApplication/Intellicus/**Web.Config file.

appSettings

You need to append information between appSettings tag start and end.

```
<appSettings>
    <add key="Intellicus Relative Path" value="/Intellicus" />
    <add key="Login Page Path" value=""/>
 </appSettings>
```

> **Important:** The relative path should start from the first folder inside the host application's folder name and end with "Intellicus". For example, if application folder structure is:
>
> C:\f1\hostapplication\f2\f3\Intellicus, then relative path would be "/f2/f3/Intellicus".
>
> If the path is not specified in this way, Intellicus DotNet Client will not be able to find config files and resources required to display the required page. As a result of this, user will receive "HTTP – 404 Page cannot be found" error.

If <appSettings> tag is not found in **/HostApplication/**Web.Config file, copy the content including start-tag and end-tag from **/HostApplication/Intellicus/**Web.Config file.

system.web

You need to append information between system.web tag start and end.

```
<system.web>
    <IntellicusUrlRewriter>
        <rule>
          <url>/core/view/.*</url>
          <rewrite>/core/CoreViewServlet.aspx</rewrite>
```

```
      </rule>
      <rule>
        <url>/InteraController.jsp</url>
        <rewrite>/InteraController.aspx</rewrite>
      </rule>
      <rule>
        <url>IntelliController.aspx</url>
        <rewrite>PrintHtmlFormat.aspx</rewrite>
      </rule>
      <rule>
        <url>/client/servlet/.*</url>
        <rewrite>/servlet/DownloadIRL.aspx</rewrite>
      </rule>
      <rule>
        <url>/common/help/images/Default/.*</url>
        <rewrite>/servlet/HelpController.aspx</rewrite>
      </rule>
      <rule>
        <url>css/images/Default</url>
        <rewrite>images/Default</rewrite>
      </rule>
      <rule>
        <url>GridWidget.jsp</url>
        <rewrite>GridWidget.aspx</rewrite>
      </rule>
      <rule>
        <url>InteraController.jsp</url>
        <rewrite>InteraController.aspx</rewrite>
      </rule>
      <rule>
        <url>/tools/amchart/templates/.*</url>
        <rewrite>/servlet/ChartTemplateController.aspx</rewrite>
      </rule>
      <rule>
        <url>/templates/grids/.*</url>
        <rewrite>/servlet/GridTemplateController.aspx</rewrite>
      </rule>
    </IntellicusUrlRewriter>
</system.web>
```

> **Important:** Delete /HostApplication/Intellicus/web.config after this
> activity is completed.

> **Important:** When host application is running under IIS7 or higher in integrated pipeline mode, Web.Config should be upgraded using "*appcmd migrate config* <website-location>" command.
>
> After migrating the Web.Config file using appcmd utility, an attribute runAllManagedModulesForAllRequests with value "true" should be added to <module> tag within <system.WebServer> tag.

Handling session timeout

When session time-out occurs, all the session information is lost and application needs to display login page.

Intellicus DotNet Client collects login information from Web.Config file. This file contains the URL to be called in event of session timeout. Until changed, this URL points to Intellicus' login page – Login.aspx. This may not be correct in an integrated environment.

You need to change Login.aspx to the host application's login page URL. Host application's login page needs to have code for redirection after successful authentication.

To change the URL, open the file Web.Config located in Intellicus folder and specify the URL of host application Login Page.

```
<add key="Login Page Path" value="/f1/Login.aspx"/>
```

Where, f1 is the directory in the host application which contains Login.aspx.

If you want to open the called page, extract the value of encryptURL query string parameter and use in the code of login page.

Specifying Intellicus logs related information

(Adding contents in Log4NetConfig.xml file)

Information about Intellicus is stored in logs. This information can be used for debugging, if required.

When logs in host application are not maintained through Log4NetConfig.xml

Copy /HostApplication/intellicus/Log4NetConfig.xml to /HostApplication folder.

In /HostApplication/Log4NetConfig.xml, replace

```
<param name="File" value="client/logs/DotNetClient.log"/>
```

with

```
<param name="File"
  value="intellicus/client/logs/DotNetClient.log"/>
```

When logs in host application are maintained through Log4Net

In /HostApplication/Log4NetConfig.xml, replace

```
<param name="File" value="client/logs/DotNetClient.log"/>
```

with

```
<param name="File"
  value="<relative file path including file name of log file>"/>
```

> **Important:** Delete /HostApplication/intellicus/Log4NetConfig.xml after this activity is completed.

Providing Access Privileges

Intellicus will need to read/write reporting specific files on the machine where IIS is installed.

Provide Full Security Access Rights either to "Everyone" user or to account name under which ASP.NET process runs on the following folders

- intellicus/client/reports

- intellicus/client/logs

- intellicus/client/temp

- Intellicus/SecureData

Setting IIS properties

ISAPI extension mapping for Intellicus

Intellicus DotNet Client uses various extensions which should be recognized by IIS. IIS needs application maps to be able to recognise the type of requests to be served.

Given below is the list of extensions to be mapped:

.htm	.properties	.jpeg
.html	.pdf	.gif
.aspx	.jsp	.bmp
.ascx	.xls	.png
.asax	.doc	.swf
.axd	.csv	.irl
.css	.jar	.ict
.js	.xml	
.config	.txt	

.class	.jpg	

Figure 9:Extensions to be mapped

> **Important:** In case host application is not using web services, above extension mappings are not required. Add .* instead as extension.
>
> When host application is running under IIS6 and individual extension mapping is specified, Create a mapping in the ASP.NET HTTP handlers configuration for each specific mapping in the IIS 6.0 scriptmaps. For more information please refer following site - http://support.microsoft.com/kb/909641
>
> When host application is running under IIS 7 integrated pipeline mode, extension mapping is not required.

Application mappings are carried out on **Application Configuration** dialog box of **Internet Service Manager**.

To open **Application Configuration** dialog box,

Open IIS interface.
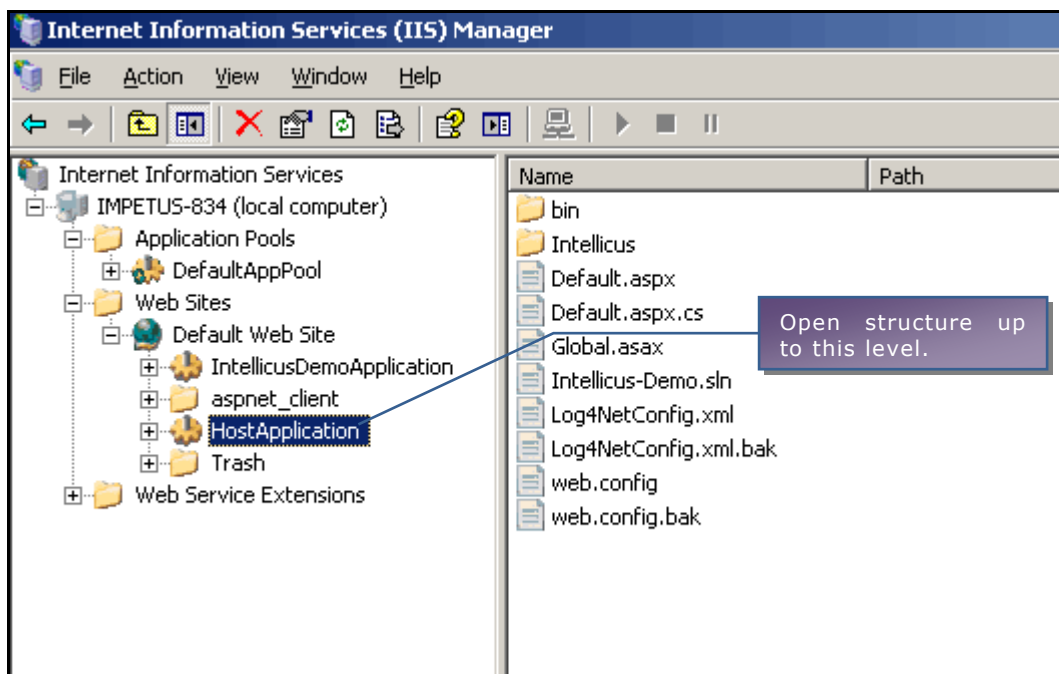
Figure 10: IIS Manager showing Host Application
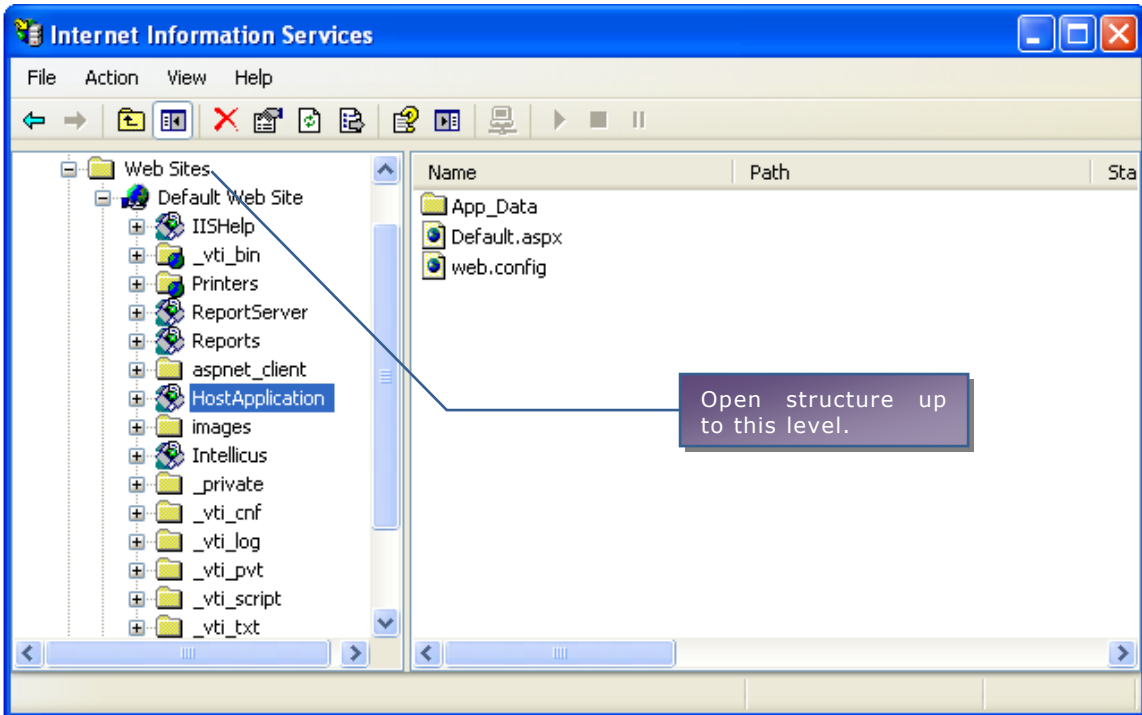
Figure 11: Host Application Folder under Web Sites Folder
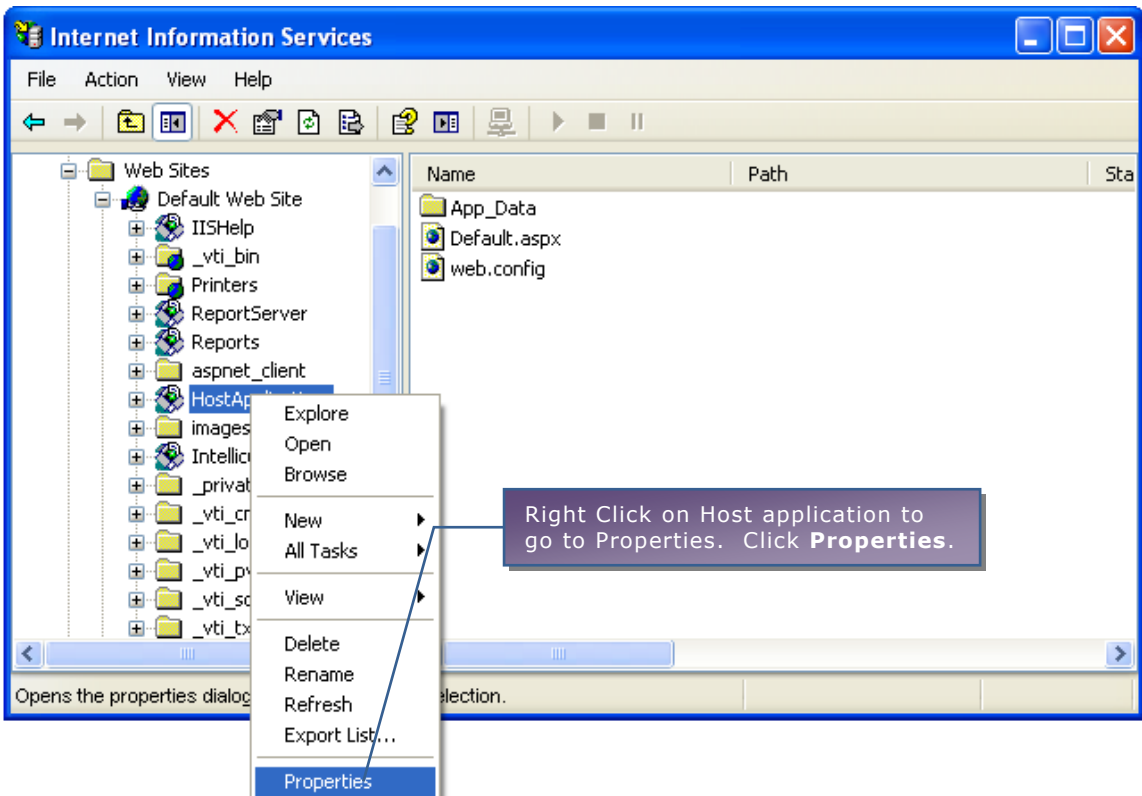
Right-click **HostApplication** node.

Figure 12: Properties of Host Application

Host Application Properties dialog box opens.

Click **Configuration…** on Directory tab's **Application Settings** area.
If **Configuration** button is disabled then click **Create** button, which is present just above the **Configuration** button.
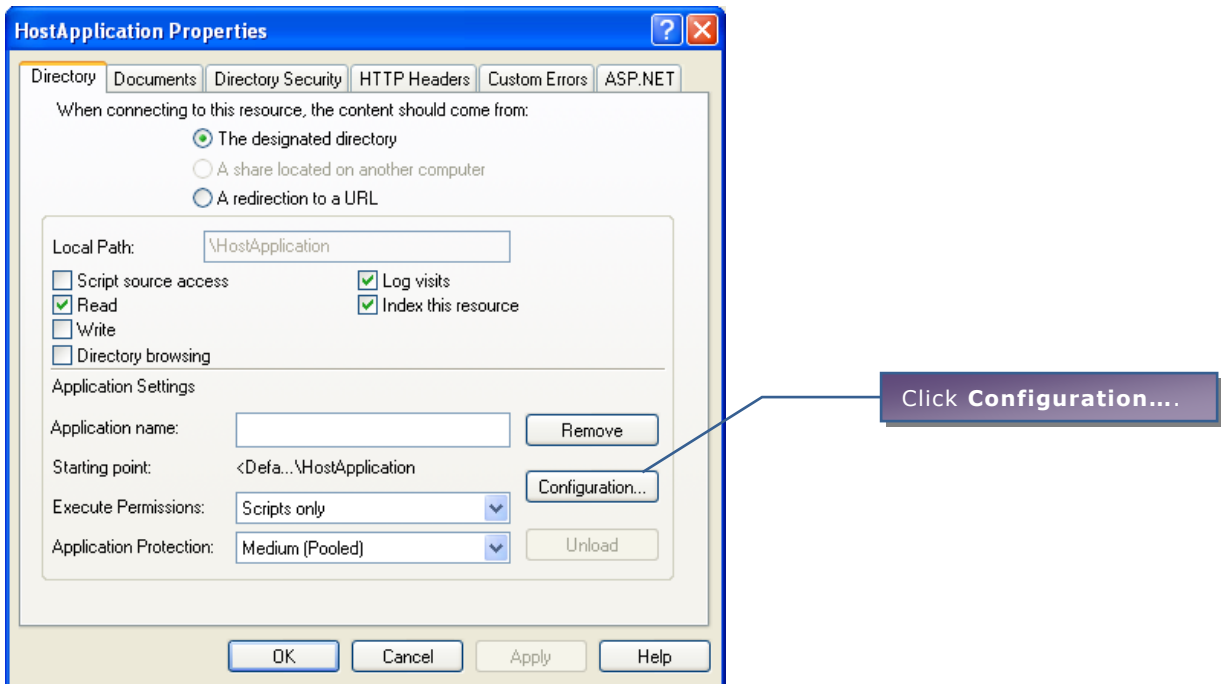

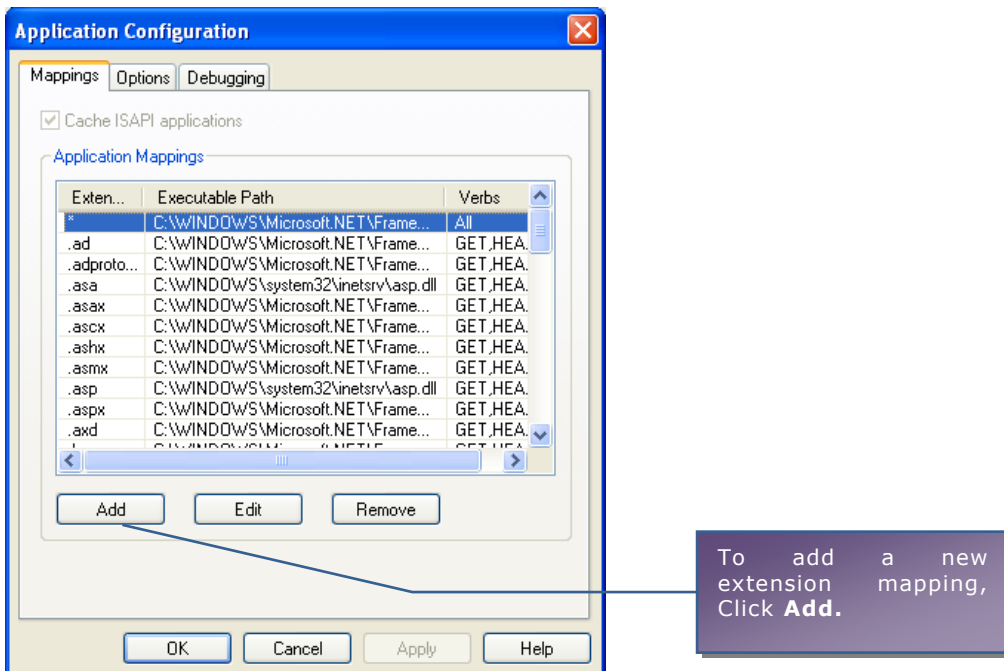Figure 13: Host Application Properties

Application Configuration dialog box appears.

Figure 14: Add an extension

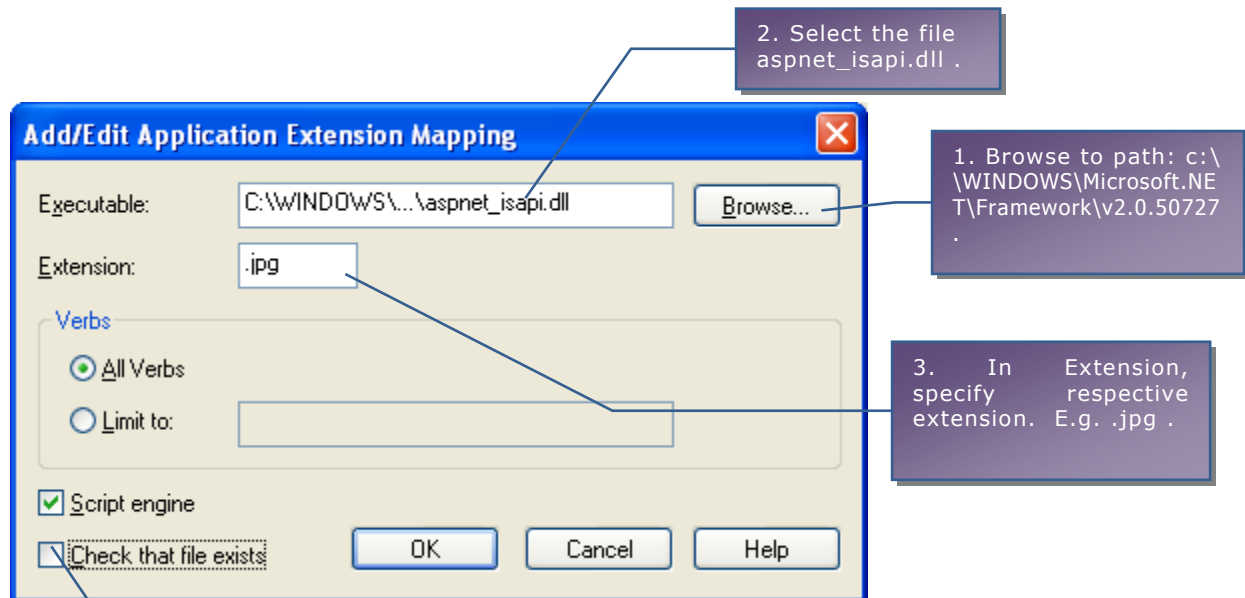**Add/Edit Application Extension Mapping** dialog box appears.

Figure 15: Application Extension Mapping

After closing **Add/Edit Application Extension Mapping** dialog, close **Application Configuration** dialog and **Properties** dialog by clicking **OK** on respective dialog boxes.

IIS needs to be restarted after having completed this step.

Restart the IIS and Close IIS interface.

Enabling Anonymous Authentication

Applet that Intellicus uses for report printing and viewing report in Jvista is unknown to IIS. Anonymous authentication needs to be enabled to load this applet.

By default, IIS has anonymous authentication enabled. In case the host application does not use anonymous authentication, it needs to be enabled for Intellicus.

To enable anonymous authentication,

On IIS Interface, right-click **HostApplication** node and on context-menu, click Properties. **Host Application Properties** dialog box opens.
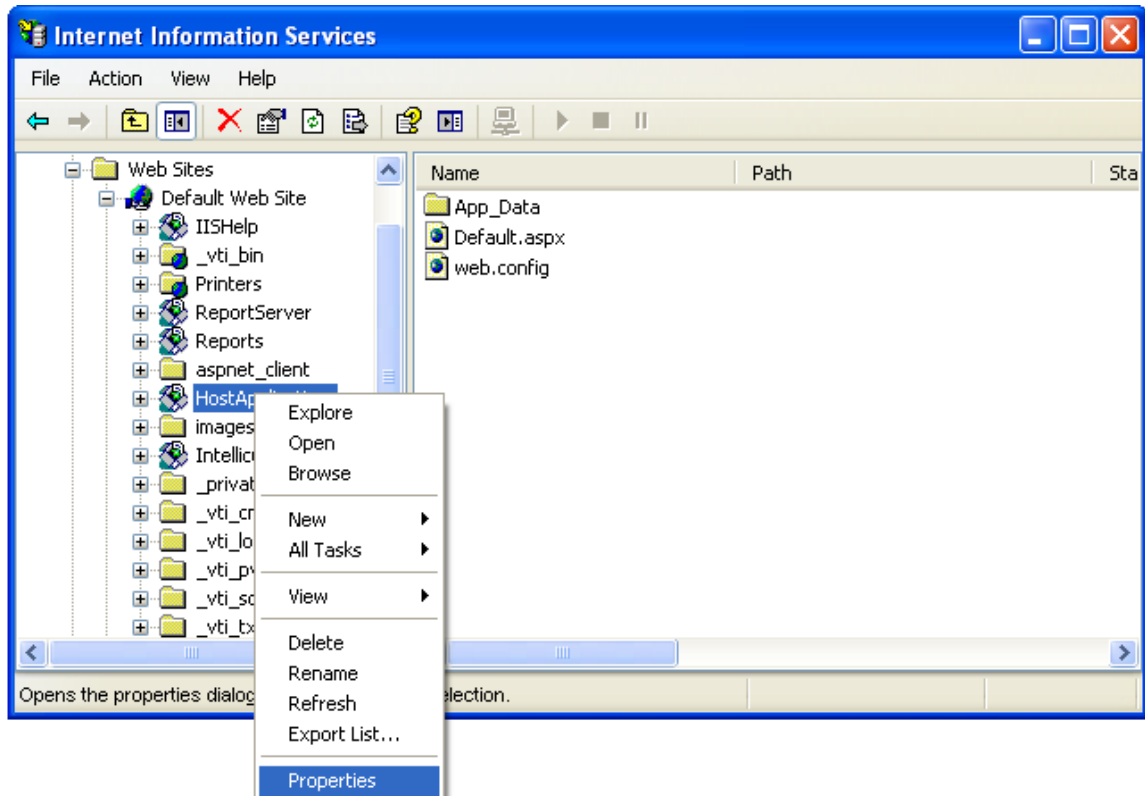
Figure 16: Host Application Properties

Select the **Directory Security** Tab and click **Edit** button of **Anonymous access and authentication control** section.
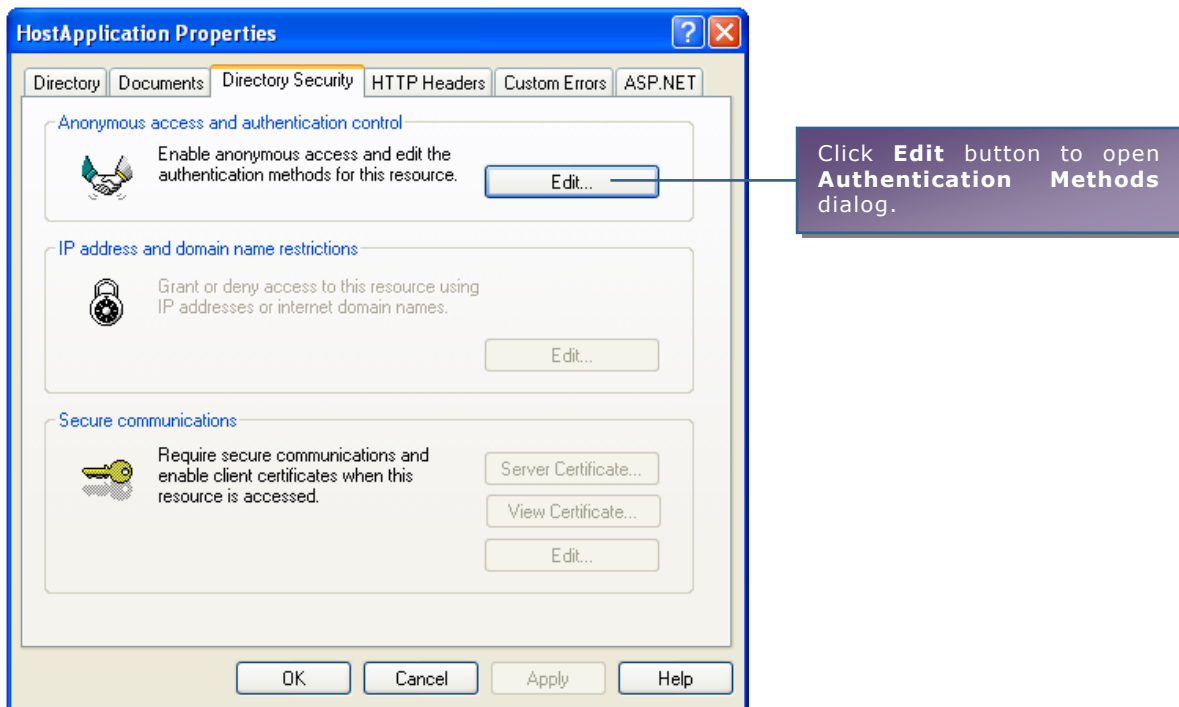

Click **Edit** button to open **Authentication Methods** dialog.

Figure 17: Check Authentication Methods

Authentication Methods dialog box opens.

If **Anonymous access** checkbox is unchecked, then follow further steps.
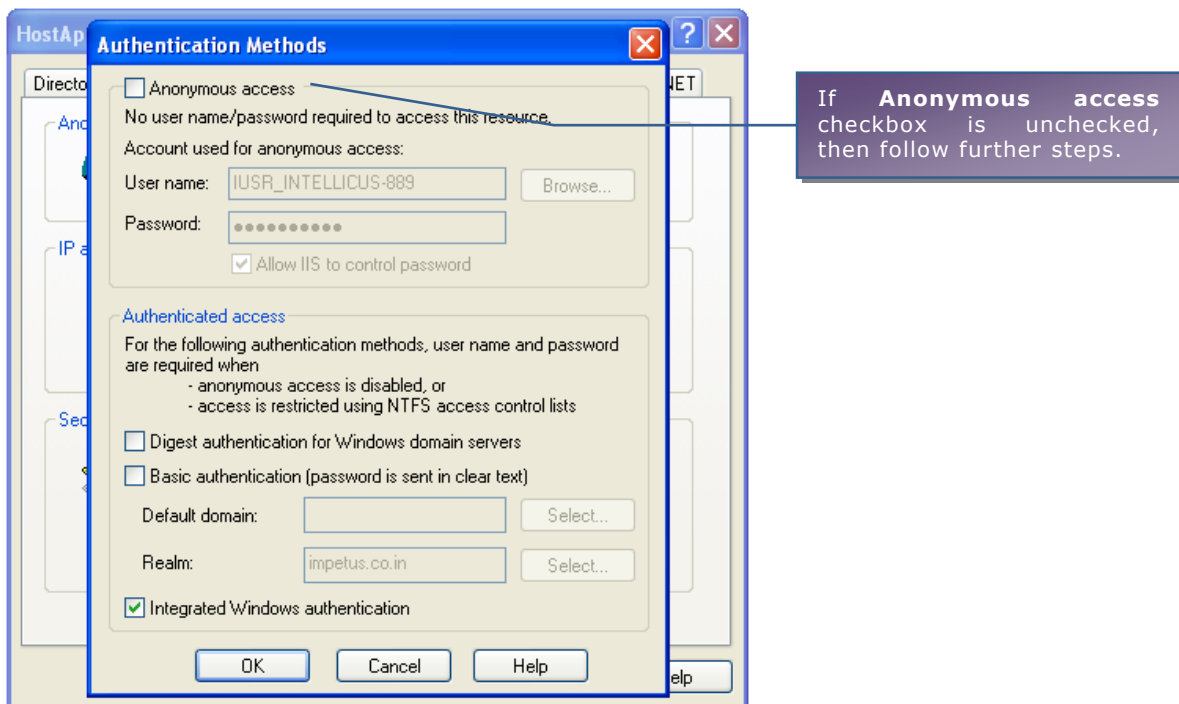
Figure 18: Authentication Methods

If Anonymous access is unchecked then follow the next steps. If it is checked, there is no need to set Anonymous access for Intellicus folder since it is already specified at root level.

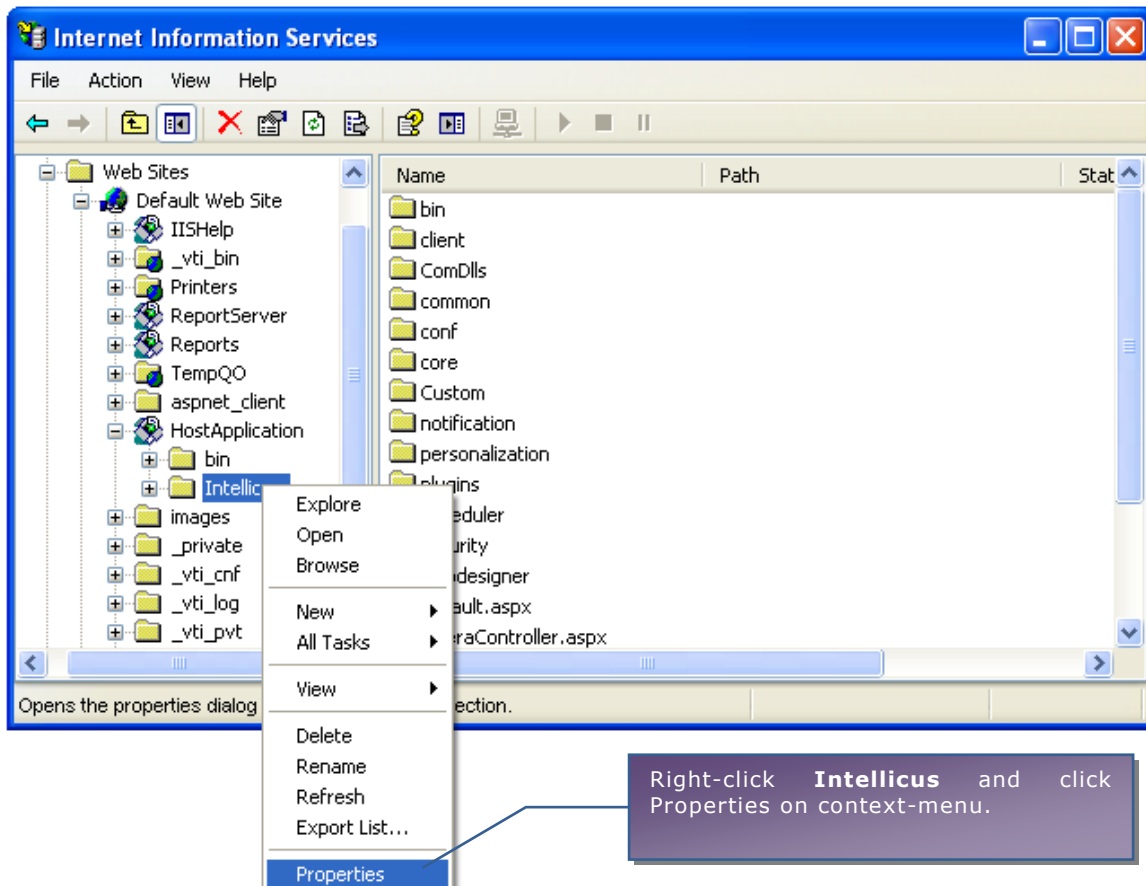On IIS interface, right-click **Intellicus** folder and on context-menu, click properties.

Figure 19: Properties of Intellicus Folder on IIS

Intellicus Properties dialog box opens. Select the **Directory Security** Tab and click on **Edit** button of **Anonymous access and authentication control** section.
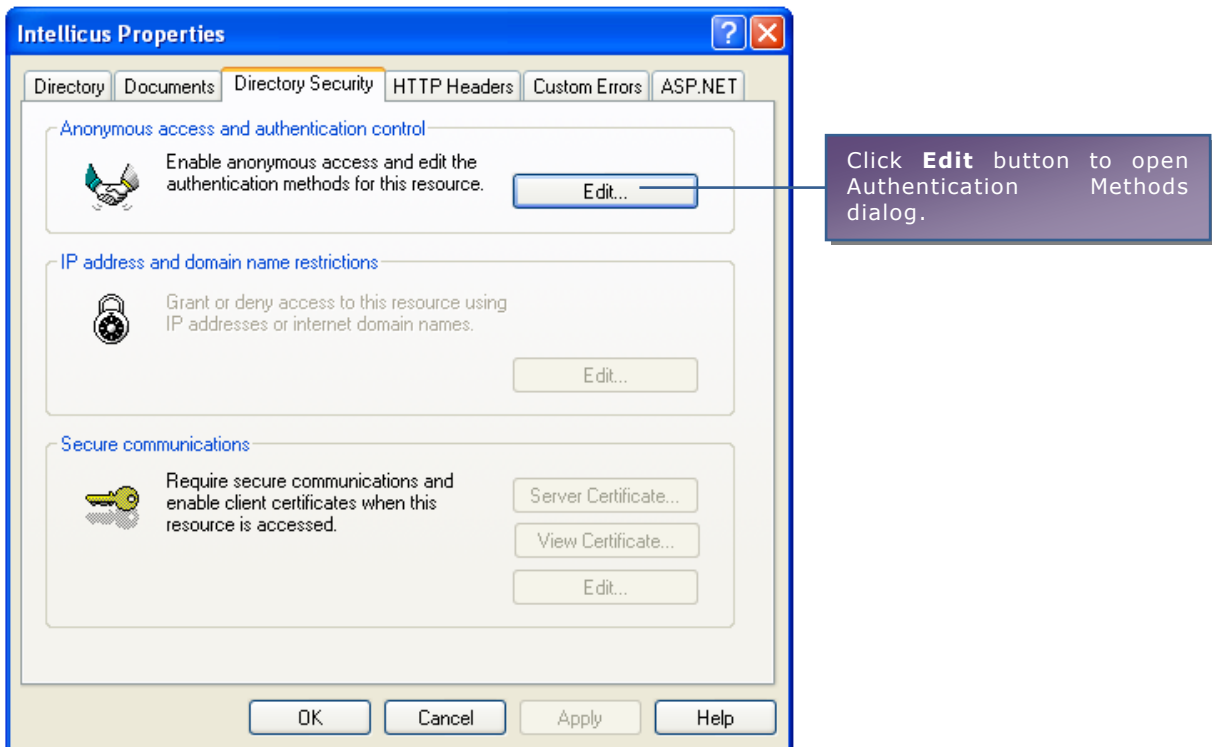

Figure 20: Intellicus Properties-Authentication Methods

Authentication Methods dialog box opens.

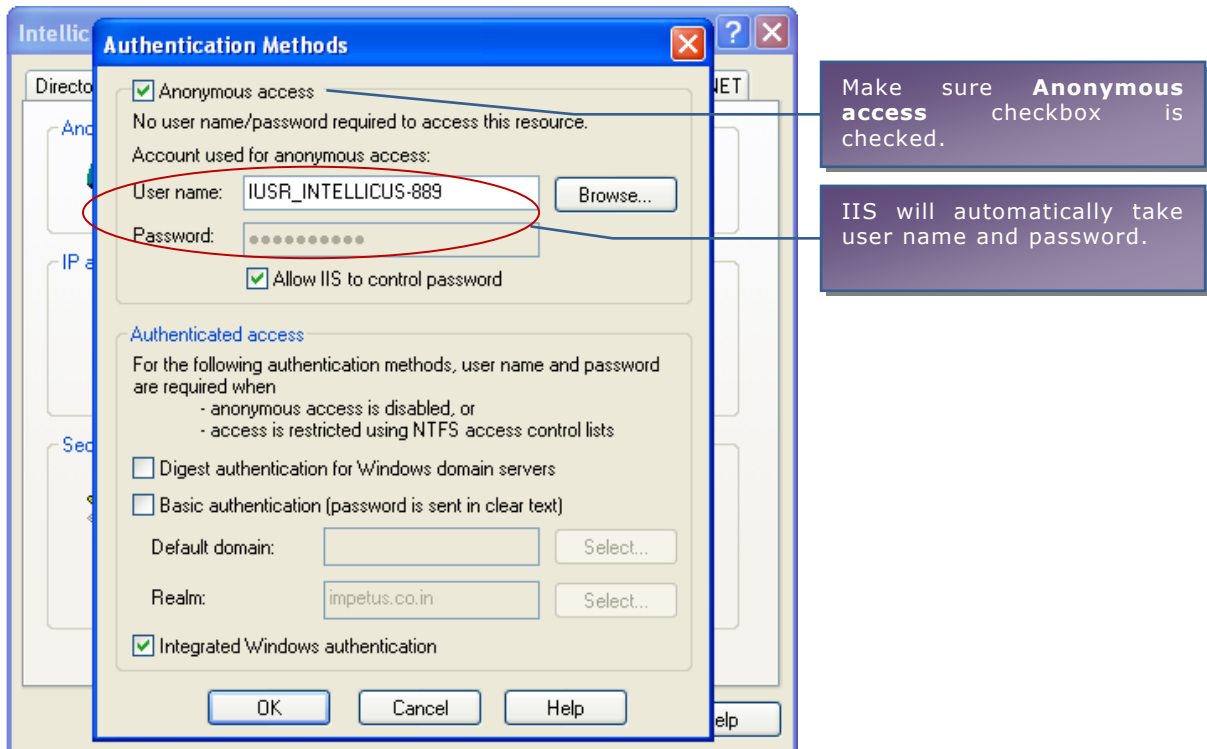On **Authentication Methods** dialog, Anonymous access checkbox needs to be checked.

Figure 21: Authentication Methods

IIS needs to be restarted after having completed this step.

Deny Access to SecureData folder

Go to the SecureData folder in IIS Manager.

Open Properties page.

Go to Directory Security tab -> Authentication and access control.

Uncheck 'Enable anonymous access' checkbox and disable all the authentication methods below.

Restart the IIS and Close IIS interface.