



## Captivate for ISA Server Documentation

(The following graphics are screen shots from Microsoft® ISA Server 2006 which is the property of Microsoft Corp. and are included here for instructive use. Some images illustrate Captivate for ISA Server, which is the property of Collective Software.)

### Table of Contents

<a href="#">Captivate for ISA Server Documentation.....</a>	<a href="#">1</a>
<a href="#">Captive portal and ISA proxy.....</a>	<a href="#">3</a>
<a href="#">Solution.....</a>	<a href="#">3</a>
<a href="#">Features.....</a>	<a href="#">3</a>
<a href="#">Requirements.....</a>	<a href="#">3</a>
<a href="#">Help is Available!.....</a>	<a href="#">3</a>
<a href="#">Installation of Captivate for ISA Server.....</a>	<a href="#">5</a>
<a href="#">Install Procedure.....</a>	<a href="#">5</a>
<a href="#">Troubleshooting.....</a>	<a href="#">5</a>
<a href="#">Install rolls back (with red error message at the end).....</a>	<a href="#">5</a>
<a href="#">Frozen or hung install.....</a>	<a href="#">5</a>
<a href="#">Setting up the Filter for a simple Agreement page.....</a>	<a href="#">7</a>
<a href="#">Main filter settings.....</a>	<a href="#">7</a>
<a href="#">Trigger settings.....</a>	<a href="#">7</a>
<a href="#">Tracking settings.....</a>	<a href="#">8</a>
<a href="#">Advanced settings.....</a>	<a href="#">8</a>
<a href="#">Policy rule settings.....</a>	<a href="#">8</a>
<a href="#">Testing the configuration.....</a>	<a href="#">9</a>
<a href="#">Changing the agreement page.....</a>	<a href="#">11</a>
<a href="#">Agreement.htm.....</a>	<a href="#">11</a>
<a href="#">Default.css.....</a>	<a href="#">11</a>
<a href="#">Background.gif.....</a>	<a href="#">11</a>
<a href="#">Other files served by Captivate.....</a>	<a href="#">11</a>
<a href="#">Replicate your file changes.....</a>	<a href="#">11</a>
<a href="#">Non-ISA-served files.....</a>	<a href="#">11</a>
<a href="#">Logging agreement events to a Windows Event Log.....</a>	<a href="#">13</a>
<a href="#">Authenticating SecureNAT users with a form.....</a>	<a href="#">14</a>
<a href="#">Prepare the Network element.....</a>	<a href="#">14</a>
<a href="#">Set up an SSL Listener.....</a>	<a href="#">14</a>
<a href="#">Create the authentication publishing rule.....</a>	<a href="#">16</a>
<a href="#">Captivate settings for the authentication rule.....</a>	<a href="#">20</a>
<a href="#">Create the proxy access rule.....</a>	<a href="#">22</a>
<a href="#">Captivate settings for the access rule.....</a>	<a href="#">24</a>
<a href="#">Authentication events in the web proxy log.....</a>	<a href="#">26</a>

<u>Creating a Captivate database.....</u>	<u>27</u>
<u>Setting up the ODBC source .....</u>	<u>28</u>
<u>Client checklist.....</u>	<u>31</u>
<u>Filter licensing.....</u>	<u>32</u>
<u>Demo/Lab mode.....</u>	<u>32</u>
<u>Troubleshooting.....</u>	<u>32</u>
<u>Support for Captivate for ISA Server.....</u>	<u>32</u>

## Captive portal and ISA proxy

There are many scenarios where a web proxy provider wishes to inject a special process before allowing access to the Internet. Some simple examples are:

- Display a “Terms of Service” screen or policy page which the user must read and acknowledge.
- On a wireless network segment, always direct the user to a custom start page first, before allowing other browsing. This could be a home page, or an external web app that collects information or payment.
- On a wireless network segment, track and log new users by IP and MAC address.
- Require users to authenticate to ISA with a web form before allowing browsing. This is useful when you cannot control the browser proxy settings, but your users will still have accounts that are known to ISA.
- Block access to other protocols (such as FTP, SSH, etc.) until a user passes the authorization process, then allow those protocols.

## Solution

*Captivate for ISA Server* from Collective Software is a filter for ISA 2006 that adds flexible captive portal functionality to your proxied networks. It directly supports all of the above scenarios, and has scriptable features to extend and enhance its functionality.

## Features

- Tight integration with ISA management; configure Captivate policies right in the HTTP access rule property pages.
- Link non-HTTP rules with a Captivate policy, so users will be blocked from that rule until they pass the Captivate authorization process.
- Expire authorizations daily or by time since last agreement.
- Can track MAC addresses instead of IPs, for hot-spot proxies where IP addresses are re-issued frequently to different computers.
- Agreement process is completely customizable with [Lua](#) scripts and modules. Use the default, build your own, or take advantage of our [expert consultants](#) to help build the perfect solution.

## Requirements

- ISA Server 2006
- Microsoft .NET Framework version 2 should be installed on each ISA server.

## Help is Available!

We are always happy to help you get our software set up and working. If you have

questions or need assistance understanding/configuring/testing a Collective product, you can get in touch with our support staff quickly and easily. For the most up-to-date information, please see our Support page at <http://www.collectivesoftware.com/Support/>

# Installation of Captivate for ISA Server

## ***Install Procedure***

1. Close the ISA management console if it's open.
2. Execute the Captivate.msi file. This will stop your firewall service, install the filter and interface software, register the filter, and then re-start the firewall service.
3. If you are installing over a remote desktop session, keep in mind that when the firewall service stops and restarts your RDP connection may be frozen, dropped or timed out. If an error occurs during the installation and the firewall service cannot be restarted, you will need to access the console to troubleshoot further (see below).
4. You must run the installer on each ISA server in an array separately, so they will all have the filter files installed and registered.
5. If the installation completes with no errors, then you can proceed to the configuration section.

## ***Troubleshooting***

The installation normally completes without errors. However there are a few possible failure modes that can occur for this complex install process.

### **Install rolls back (with red error message at the end)**

If you are presented with an error message on the final screen, then check out the application event log, which often will contain details on why the installation failed. The problem may be immediately solvable from this information, or you may need to work with Collective support for additional troubleshooting assistance.

### **Frozen or hung install**

The installer tries to start the firewall service after it is done registering the filter components. In rare cases, everything may register properly but there could still be a problem preventing the firewall service from starting. In this situation, the installation may appear to hang on the "Starting services..." item. This is because it is trying repeatedly to start the service, and failing. In fact if you look at the application event log, you will see several errors from the firewall service as it tries to start. These messages may help identify the cause of the problem.

The install should eventually give up on starting the service, but it may take a long time. If necessary, you can expedite the rollback by going into the services control panel and setting the Microsoft Firewall service to Disabled temporarily (and applying that change). This will cause the installer to quickly give up, and it should then correctly roll back the installation while leaving the firewall service down. After this happens you can then re-enable and restart the firewall service.

This kind of problem should not normally occur, and will probably require additional troubleshooting by Collective support. However if you are able to fix the problem you

can re-run the install safely after completing this procedure.

# Setting up the Filter for a simple Agreement page

## Main filter settings

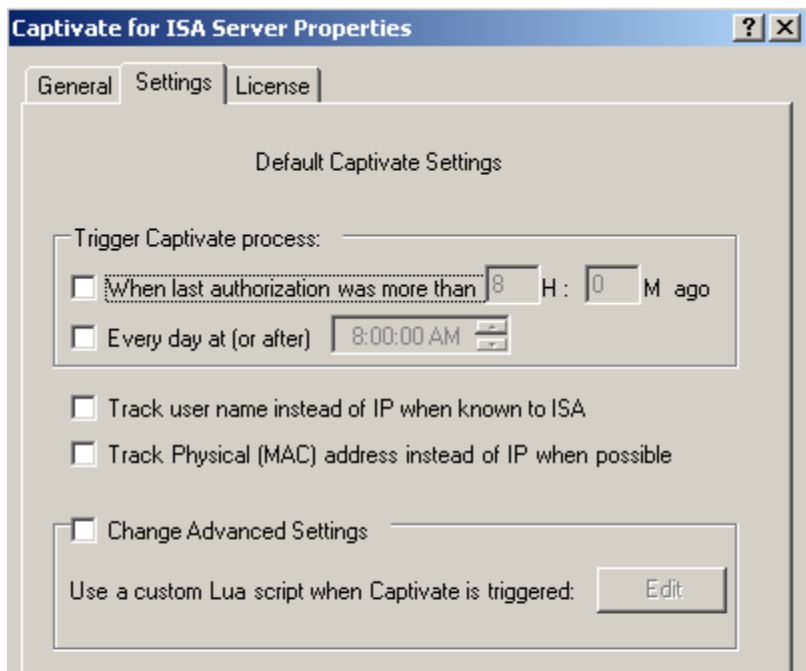
The main settings for Captivate can be found in the Web Filter properties. In the Add-ins section, choose the Web Filters tab:



right-click Captivate and go to properties:



Choose the settings tab:



## Trigger settings

By default, the Captivate process has no triggers, so even if you enable Captivate to run for an HTTP rule, no agreement process will be run.

Select one or both of the trigger options.

- *When last authorization was more than [#:##] ago*: Trigger the agreement process every time this amount of time has passed since a user, IP, or MAC address last agreed.
- *Every day at (or after) [time]*: This is designed to show the agreement once per day to each user. You can set the time of day that the filter will consider to be the start of a “new day” and expire agreements from the previous day.

If you select both options, the agreement will be run daily **and** also once per interval you choose.

## Tracking settings

By default, Captivate tracks agreements by IP address. If your access or publishing rule requires authentication, then you can select the *Track User name* option. This way, if a user logs in from several computers, they will not have to go through the agreement process several times. If you choose this option but user names are not available, then IP addresses will be tracked instead.

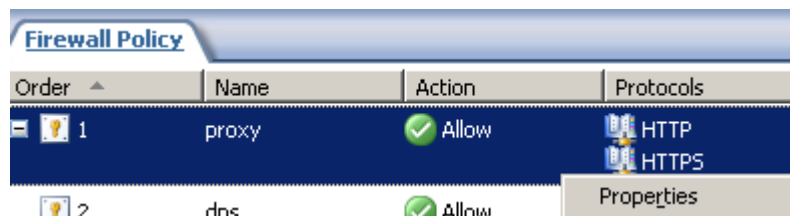
If users are configured to use ISA as their default gateway, ISA should be able to track physical MAC addresses for all connected parties. You can select the *Track MAC address* option if your network recycles IP addresses frequently (such as in a hot-spot configuration). This option should **not** be used when there are one or more routers between the client machines and ISA; you will just get the address of the nearest router instead of the actual client MAC.

## Advanced settings

The action to perform when the agreement process is run can be completely changed by replacing the default Lua script. This will be explored in below sections. The default processing logic is to show an agreement page that must be submitted before the agreement process is considered “authorized”. You can examine how this works by using notepad to open the file “[ISA Folder]\Collective Software\Captivate\TryAuthorize.lua”. This is the logic used by default if the advanced settings are not configured. The meaning of this code is discussed further in a future section.

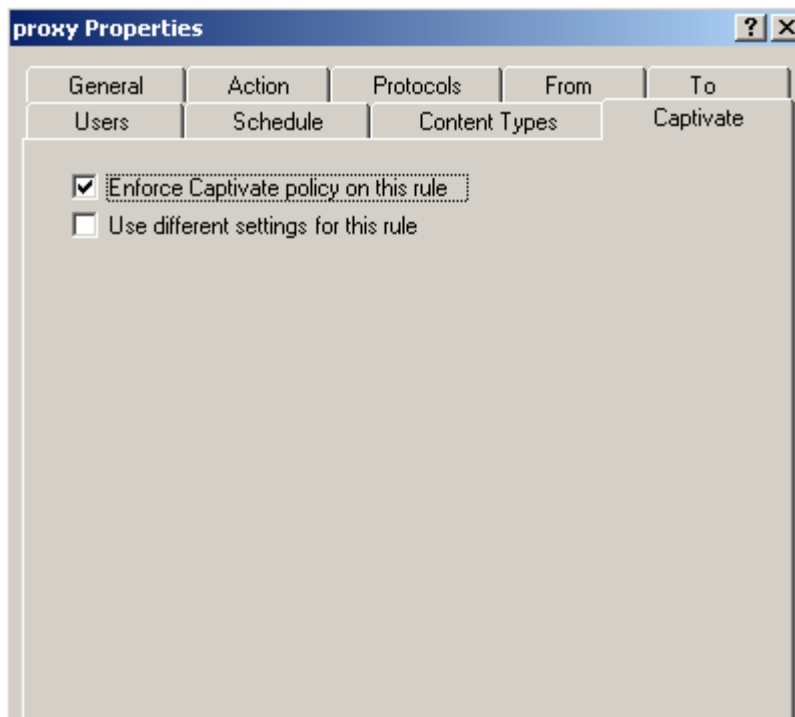
## Policy rule settings

For an agreement page to be shown, Captivate must be enabled for the firewall rule that will carry the users' web proxy traffic. Find the HTTP proxy rule you wish to run Captivate with, and go into its properties tab:



Select the Captivate tab:

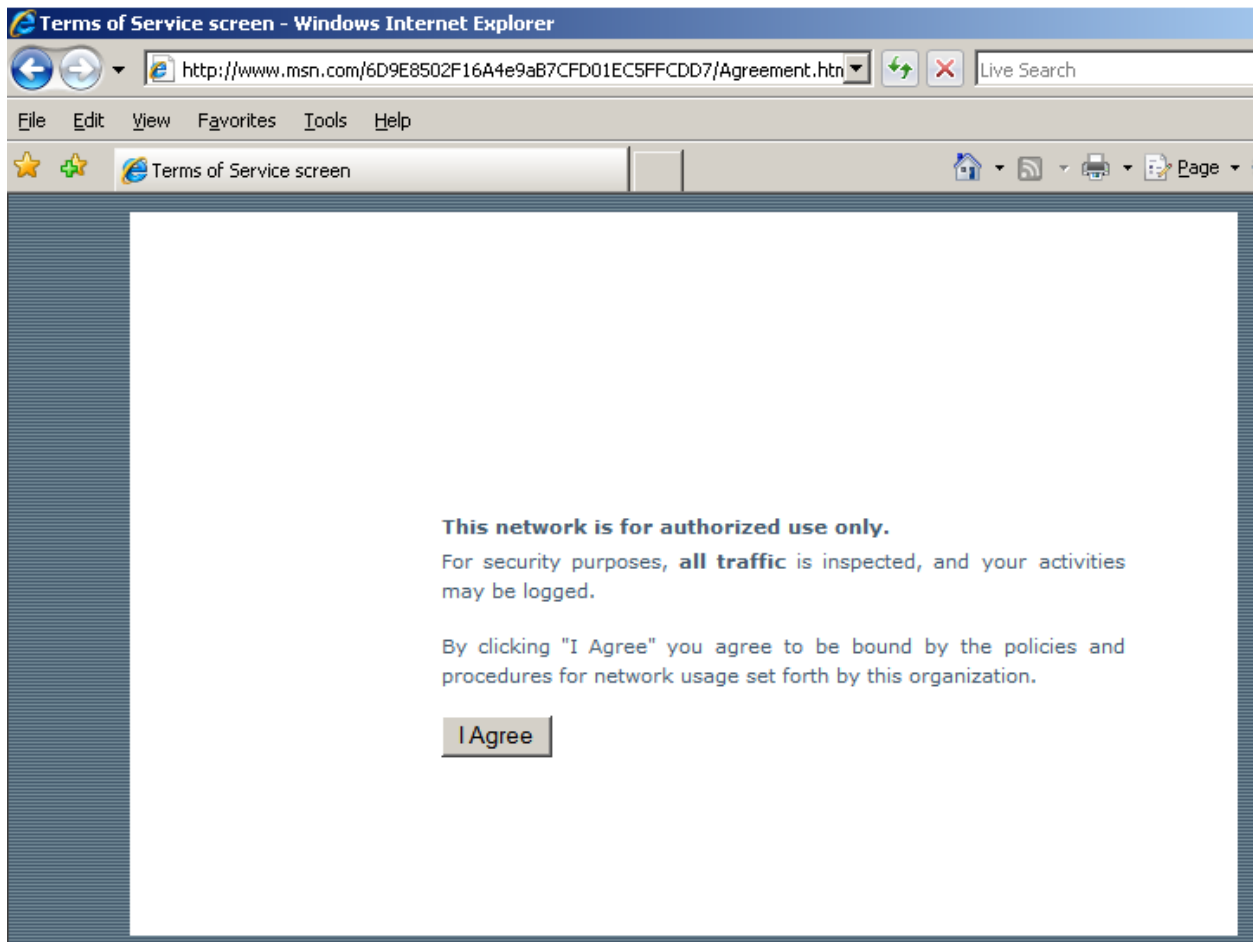




- *Enforce Captivate policy on this rule*: Select this to make this rule participate in the agreement process you configured in the main filter settings tab.
- *Use different settings for this rule*: If you choose this option, the dialog will show a full set of options to configure. This is useful when a particular rule must follow a different process than the default one configured in the main filter settings tab. The settings in this dialog will be used *instead* of the main filter options. We will use this option in the authentication example later, but it is not needed here.

### ***Testing the configuration***

After applying these settings, it should be possible to verify the agreement process is working correctly. Open a new browser whose traffic will flow through the above configured rule, and try to navigate to a URL. You should be directed to a simple agreement page (we will see how to customize it next):



Clicking "I Agree" should forward you to your requested page:



## Changing the agreement page

All HTML files, images, CSS, and scripts served by the filter are in the folder `[ISA Folder]\Collective Software\Captivate\HTMLFiles`, and you can change them and make new ones. Note that you cannot put server-side scripts such as ASP or ASPX here. ISA isn't a full web server and does not have the ability to interpret these files.

### Agreement.htm

If you use the default Lua logic as we have done in the example above, the user is redirected to a page called "Agreement.htm", which gets served from this folder. It is a very simple page that imports a style sheet and contains a form with the "I Agree" button. You can change the contents this page by editing the file. A copy of the original file is also retained as "Agreement\_example.htm", which is not used by the filter.

### Default.css

This is a simple style sheet that loads the background image and sets the look and feel for the agreement page. You can change this file to make the styles look different. If you replace the "import" directive in the agreement.htm file, then this file won't be loaded at all.

### Background.gif

This is the image loaded by default.css. You can replace it with your own, or use html or css directives to specify other image files.

### Other files served by Captivate

You can make and refer to other image, style, or javascript files in your agreement page and host them on ISA. All files **must be in the HTMLFiles** folder. For security reasons, the filter will *only* look in this folder when serving files.

### Replicate your file changes

If you use ISA Enterprise edition, all files you change or create here must be copied to the HTMLFiles folder on all members of the array. Otherwise you will get inconsistent or erroneous behavior.

### Non-ISA-served files

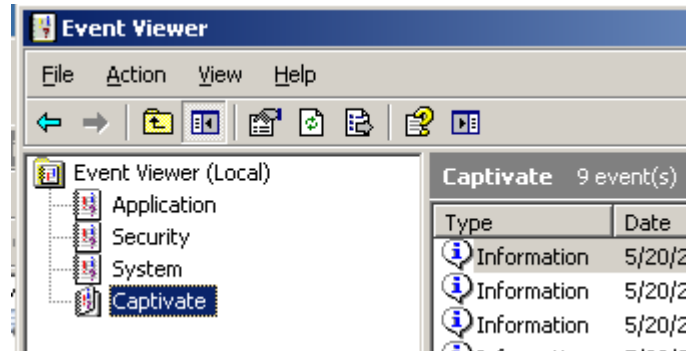
If you use *absolute URLs* in the agreement.htm or other files, then the browser will try to request them. You should make a separate firewall rule matching these requests, and place it *above* the Captivate'd proxy rule in ordering. This higher rule should *not* be set to use Captivate. Otherwise, the requests for these resources would just get blocked by Captivate (since the user hasn't agreed yet!)

The Captivate file serving code is not as efficient as a real web server, so for large files (or if you find the process is too slow), hosting image/css/script files on a real web server may yield much faster performance. As noted above, you should make a non-Captivate'd HTTP access rule to match these external requests.

If you use different Lua script logic for your agreement process, then these ISA-served files may not be needed at all. You could serve all agreement files from an IIS server, for example.

## Logging agreement events to a Windows Event Log

Upon installation, Captivate sets up a custom event source, which you can observe in Event viewer:



This log is for custom use, by default no events are issued to it (errors and ISA events are recorded in the default "Application" log).

You can utilize the custom event log and record any information you wish, by using the *LogEvent* script function.

If you wish to record agreement events, an example script is included in the installation folder, under the path "Program Files\Microsoft ISA Server\lua\examples\EventLog.lua". Paste the contents of this file into the advanced lua script dialog, and it will be used instead of the default behavior. See the script reference for further documentation of the *LogEvent* function, and related items.

Note: Sometimes the event log service cannot immediately create the new log file. If you see agreement events appearing in the Application log instead, and/or it appears that the Captivate log contains the same entries as the App log, reboot the server and try again.

## Authenticating SecureNAT users with a form

In scenarios where it is not possible or desired to have clients on a network configure web proxy settings, but authentication is still desired, Captivate can be configured to provide a solution.

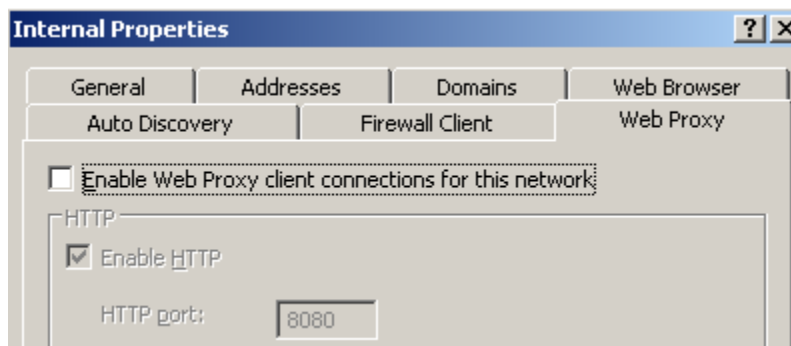
Without proxy settings, there is no true way to maintain an authenticated session, so the client's IP address is the only way to correlate activity to an individual. First, we will use Captivate to redirect users to the ISA FBA page, so they can be authenticated. Then (optionally), a record matching the user to their IP will be logged to an ODBC database, for example the SQL instance used for ISA log events.

Even though ISA's logs will mainly show anonymous IP addresses, each act of authentication will log a single line to the web proxy log containing the user name and IP address. This data can be used to correlate user names and times to the IP addresses.

There are several steps needed to achieve this setup, as detailed in the following sections.

### ***Network elements and authentication settings***

You must choose which network element will have the SecureNAT clients on it that you wish to authenticate. It is a best practice that the web proxy listener not be enabled on this network, because clients that attempt to use proxy settings on it may fail (see below). In our example we use the default Internal network for our clients:



It is possible to use both Proxy-mode connections *and* authenticated SecureNAT clients on the same network element, but you will **have to use separate source/target sets** in the "From"/"To" portions of your firewall rules. ISA cannot understand the possibility of having both SecureNAT and Proxy connections allowed between the same source and destination; it always chooses the first applicable rule and skips the others. (E.g. if you have an anonymous rule and an authenticating rule that could both match, *only the first one* will be tried.) Although it seems intuitive that ISA should first try anonymous rules and then try authenticating next, that is not how it works! This is an ISA limitation that is impossible to change.

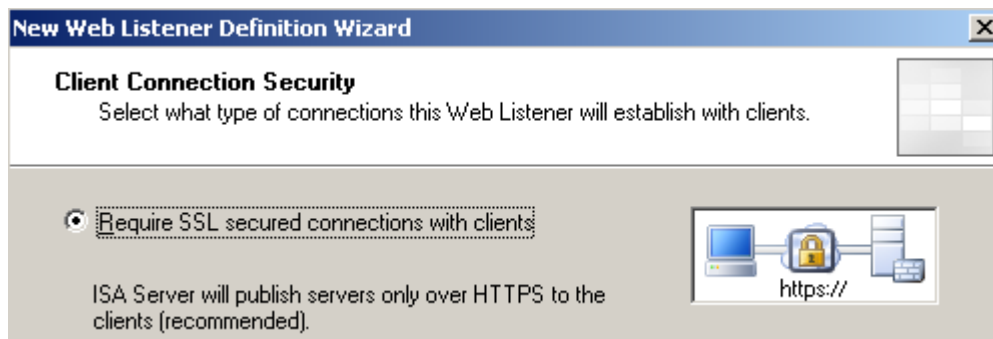
If you leave the Web Proxy listener selected on your network, be aware that Proxy-mode traffic match a Captivate rule will fail, because it will try to loop through the Captivate authentication listener (defined below).

## Set up an SSL Listener

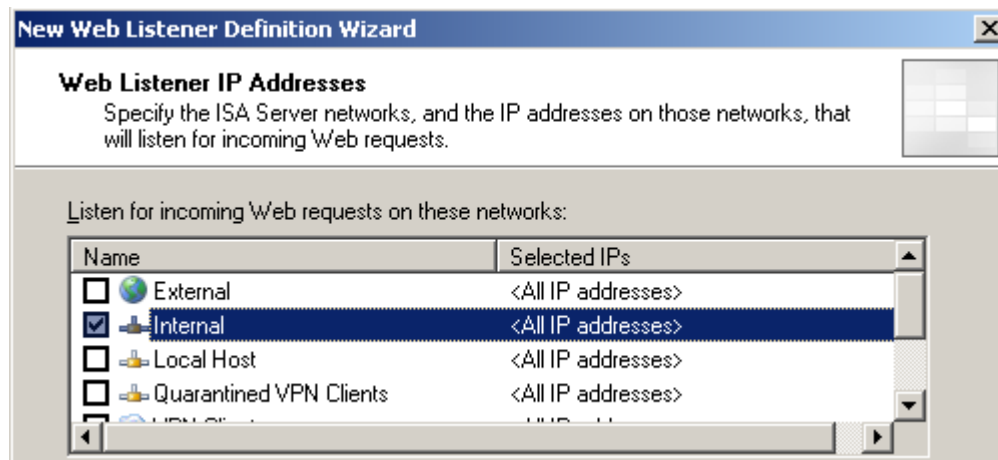
The first step to using an FBA is to have a web listener that can serve it.



It should use SSL to protect user credentials from eavesdropping attacks on the network.

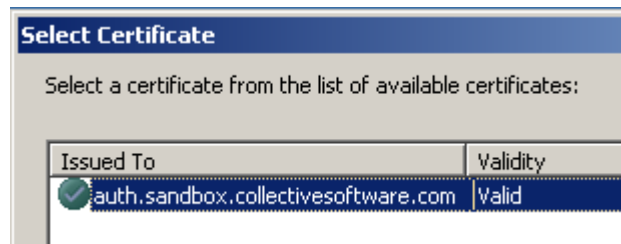


We are going to authenticate users on the Internal network:

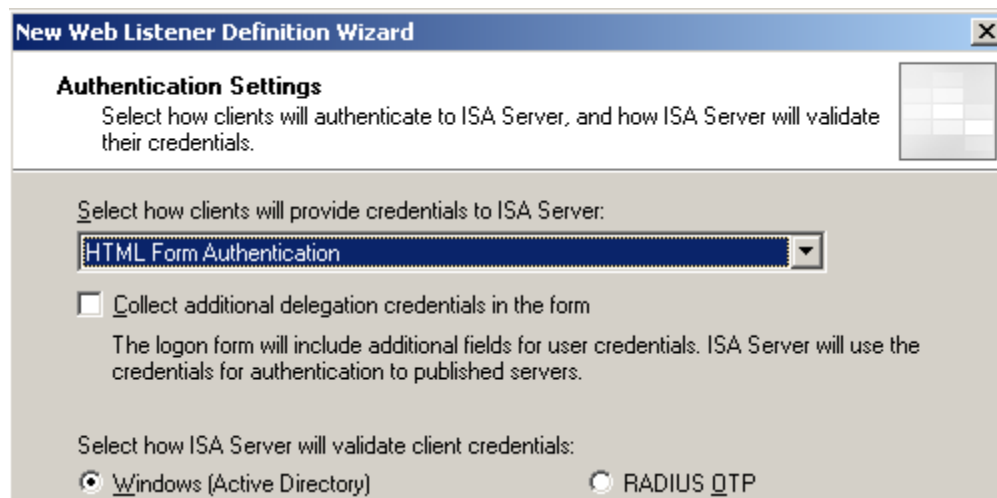


The certificate you select is important. The subject name ("Issued to" name) is where clients will be directed for authentication. It should be issued by an authority that your

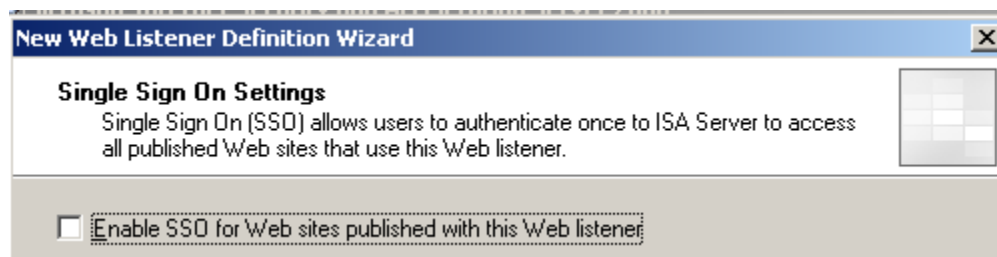
web clients will trust by default, so there is no security warning. Furthermore, the clients' DNS settings should resolve this name to the Internal IP address of the ISA server, so that they can reach the listener.



We will use Form authentication, with Active Directory:



SSO is not leveraged in this configuration:



### **Create the authentication publishing rule**

We need to publish something with that listener in order to get the FBA to appear, so create a web publishing rule:



**New Web Publishing Rule Wizard**

Microsoft  
**Internet Security & Acceleration Server 2006**

## Welcome to the New Web Publishing Rule Wizard

This wizard help you publish Web sites. Web publishing rules match incoming client requests to the appropriate Web site on the Web server or Web farm.

Web publishing rule name:

**New Web Publishing Rule Wizard** [X]

### Select Rule Action

Specify how you want this rule to respond when the rule conditions are met.

Action to take when rule conditions are met:

**Allow**

With this option selected, incoming requests matching the rule conditions will be allowed.

**New Web Publishing Rule Wizard** [X]

### Publishing Type

Select if this rule will publish a single Web site or external load balancer, a Web server farm, or multiple Web sites.

**Publish a single Web site or load balancer**

Use this option to publish a single Web site, or to publish a load balancer in front of several servers.


**New Web Publishing Rule Wizard** [X]

### Server Connection Security

Choose the type of connections ISA Server will establish with the published Web server or server farm.

**Use SSL to connect to the published Web server or server farm**

ISA Server will connect to the published Web server or server farm using HTTPS (recommended).



This next step is unusual. We aren't actually going to be publishing a real web server,

we just need to fill in something that the rule can use. Set it to the Internal IP address of ISA, just so there's no delay to look up the fake name:

**New Web Publishing Rule Wizard** [X]

**Internal Publishing Details**  
Specify the internal name of the Web site you are publishing.

The internal site name is the name of the Web site you are publishing as it appears internally. Typically, this is the name internal users type into their browsers to reach the Web site.

Internal site name:

ISA Server may not be able to connect to the server hosting the published Web site unless its computer name or IP address is specified. For example, the computer name or IP address must be specified if ISA Server cannot resolve the internal site name.

Use a computer name or IP address to connect to the published server

Computer name or IP address:

**New Web Publishing Rule Wizard** [X]

**Internal Publishing Details**  
Specify the internal path and publishing options of the published Web site. You can publish the entire Web site, or limit access to a specified folder.

Enter the name of the file or folder you want to publish. To include all files and subfolders within a folder use /\*. Example: folder/\*.

Path (optional):

Based on your selection, the following Web site will be published:

Web site:

Forward the original host header instead of the actual one specified in the Internal site name field on the previous page

The Public Name should match the certificate name:

**New Web Publishing Rule Wizard**

**Public Name Details**  
Specify the public domain name (FQDN) or IP address users will type to reach the published site.

Accept requests for:  ▼

Only requests for this public name or IP address will be forwarded to the published site.

Public name:   
Example: www.contoso.com

Path (optional):

Select the listener created above:

**New Web Publishing Rule Wizard**

**Select Web Listener**  
The Web listener specifies the IP addresses and port on which the ISA Server computer listens for incoming Web requests.

Web listener:  ▼

Listener properties:

Property	Value
Description	
Networks	Internal
Port(HTTP)	Disabled
Port(HTTPS)	443
Certificate	auth.sandbox.collectivesoftware.com
Authentication methods	FBA with AD

Delegation is not needed since this rule is just to do authentication and won't actually be publishing anything:

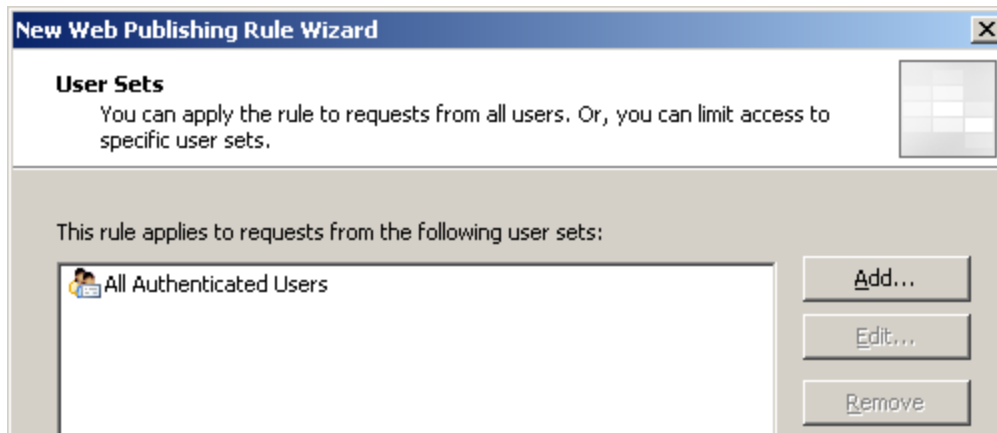
**New Web Publishing Rule Wizard**

**Authentication Delegation**  
Authentication delegation is the method ISA Server uses to authenticate the session it opens with the published site.

Select the method used by ISA Server to authenticate to the published Web server:

▼

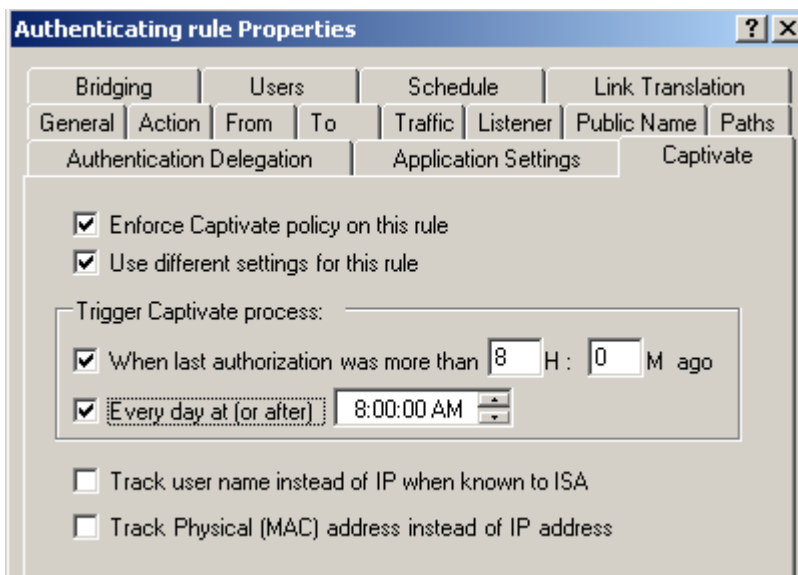
Select some appropriate users or groups; minimally you can choose "Authenticated Users" to allow anyone known to AD or ISA:



Finish the wizard and *Apply* the changes at this stage.

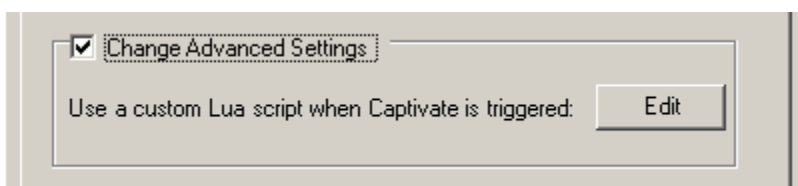
### ***Captive settings for the authentication rule***

Go into the Properties of the publishing rule just created, and select the Captivate tab. Fill it out as below. You can substitute different trigger intervals than the ones shown. These times determine how often to show the authentication screen for a given IP address.

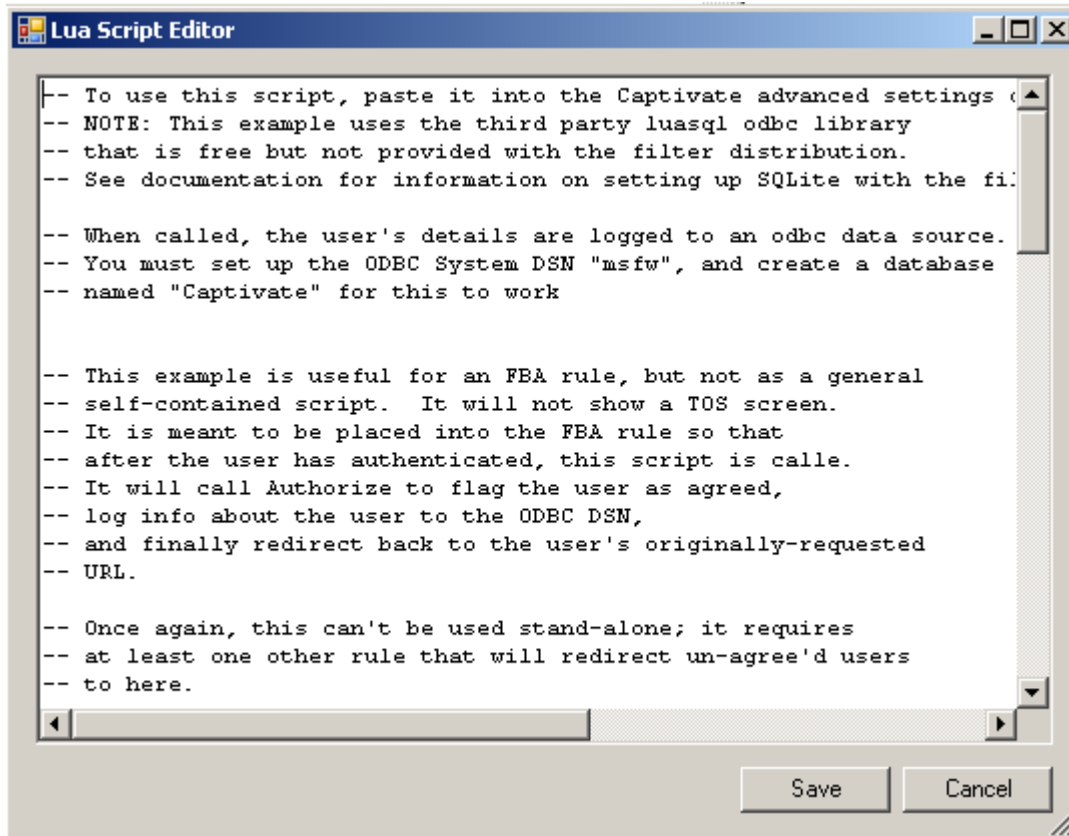


Next, we need to set up the behavior of the filter for this rule. Once the user is authenticated by the FBA, we want to (optionally) write a row into the database, then direct the user back to their originally requested URL. There is an included lua script to do this: [ISAFolder]\Collective Software\Captivate\lua\examples\LogOdbc.lua.

Select "Change Advanced Settings" and Edit:



and then paste the contents of LogOdbc.lua into the editor window:



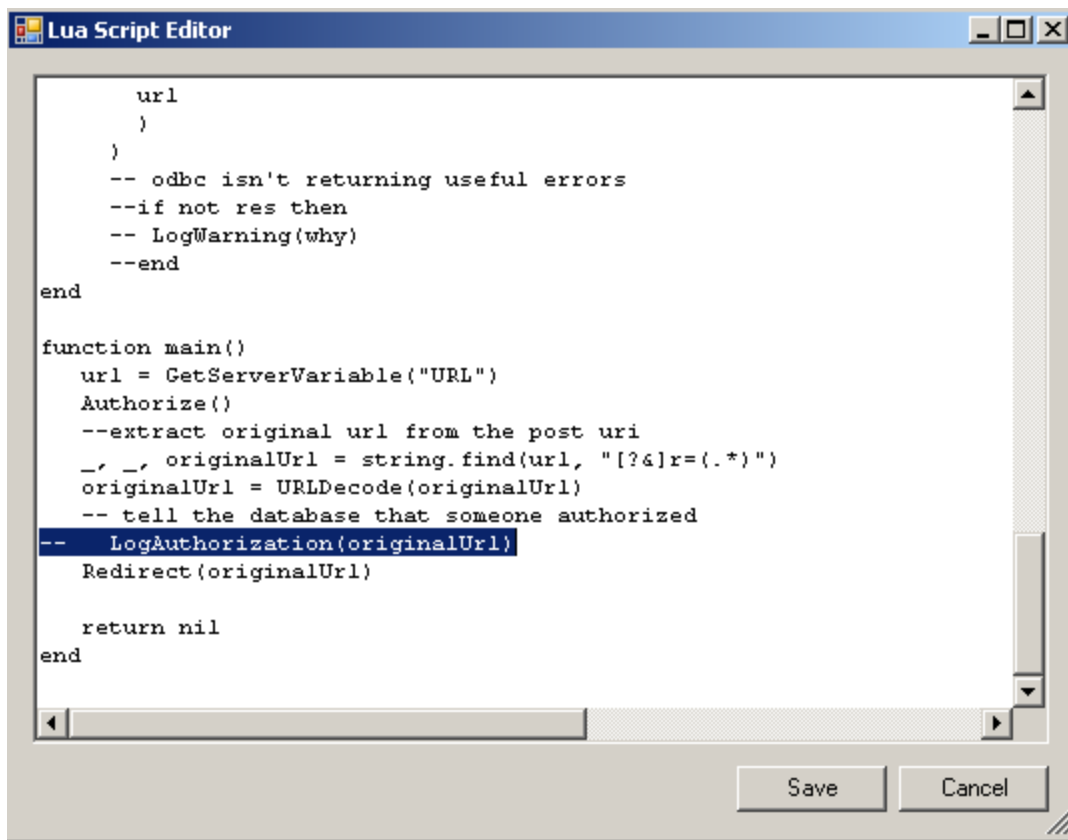
```
-- To use this script, paste it into the Captivate advanced settings
-- NOTE: This example uses the third party luasql odbc library
-- that is free but not provided with the filter distribution.
-- See documentation for information on setting up SQLite with the fi:

-- When called, the user's details are logged to an odbc data source.
-- You must set up the ODBC System DSN "msfw", and create a database
-- named "Captivate" for this to work

-- This example is useful for an FBA rule, but not as a general
-- self-contained script.  It will not show a TOS screen.
-- It is meant to be placed into the FBA rule so that
-- after the user has authenticated, this script is calle.
-- It will call Authorize to flag the user as agreed,
-- log info about the user to the ODBC DSN,
-- and finally redirect back to the user's originally-requested
-- URL.

-- Once again, this can't be used stand-alone; it requires
-- at least one other rule that will redirect un-agree'd users
-- to here.
```

**Important:** If you are **not** planning to log authentications to a separate (ODBC) data source, you should comment out the call to “LogAuthentication” near the bottom of the script, by adding two “minus” (dash) characters before it, as shown below. Even with this line commented out, a row for authentication will still be sent to the ISA web proxy log.

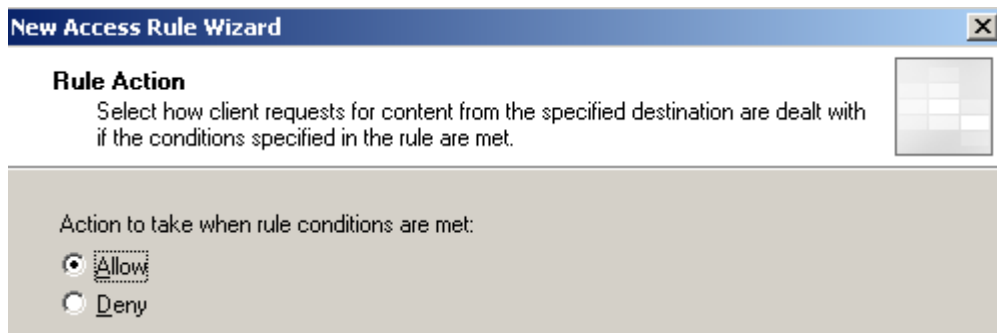


Save and *Apply* these changes.

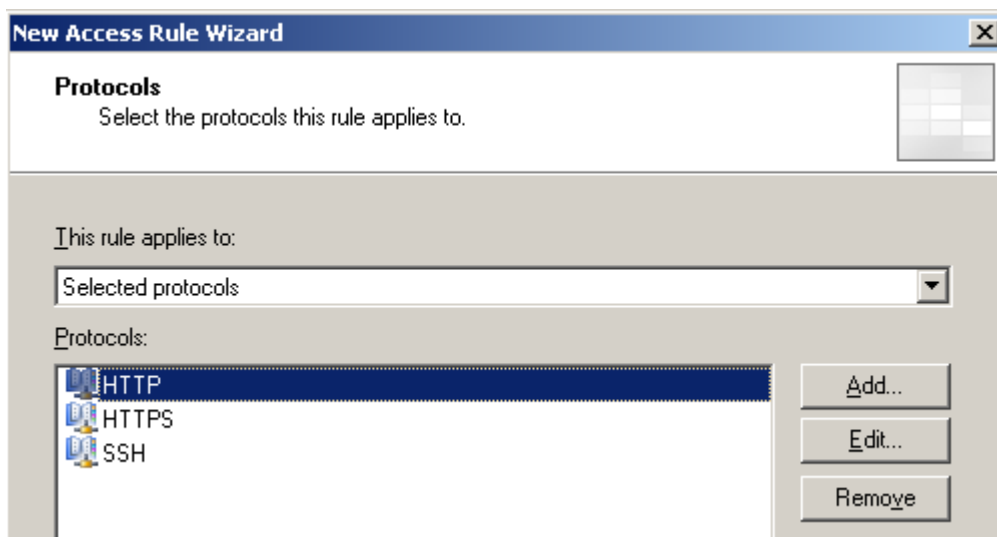
### ***Create the proxy access rule***

Next, we need to create the access rule that our clients will really be using to get from the Internal network to the Internet. Your rule may not be exactly like this example; you may allow different protocols or restrict the source or destination set.

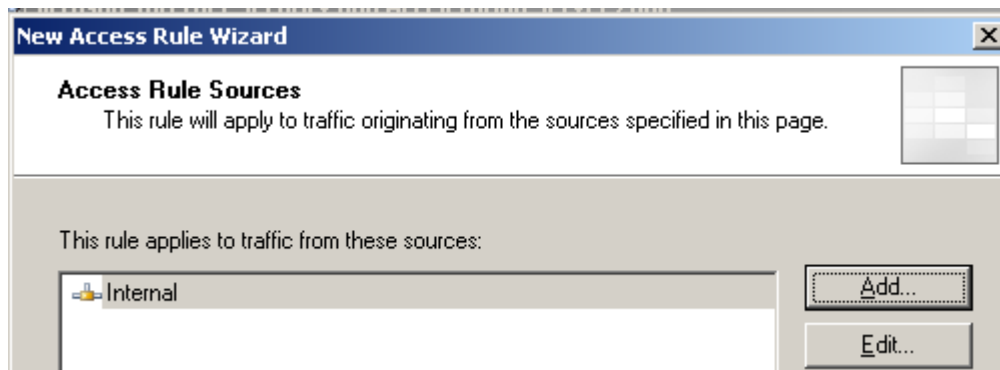




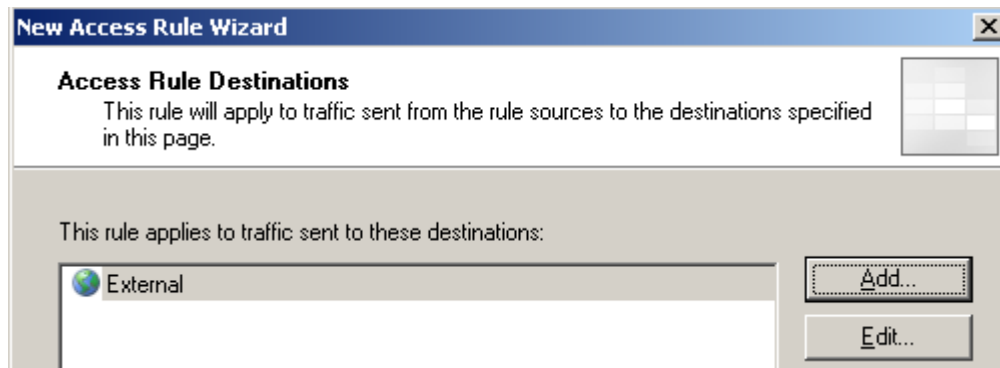
Be sure to select at least HTTP, since that's the protocol we'll use to redirect the user to the FBA. Here we are also allowing HTTPS and SSH. Below we'll see how to restrict these non-HTTP protocols so they cannot be accessed through this rule until the user has authenticated.



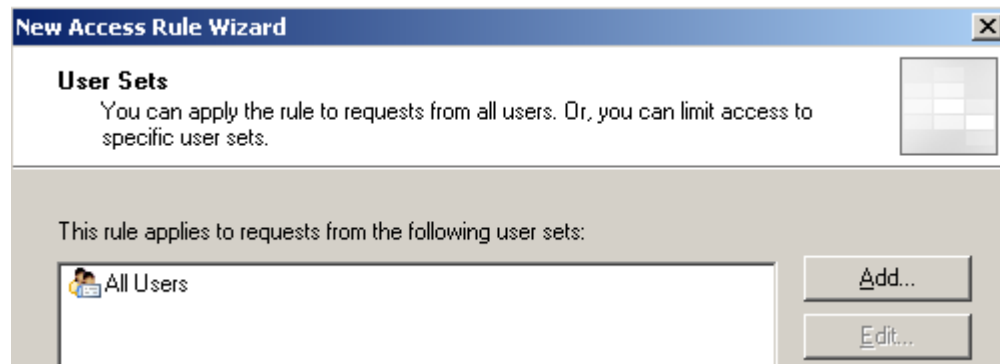
Select the Internal network being used for the SecureNAT clients as the source:



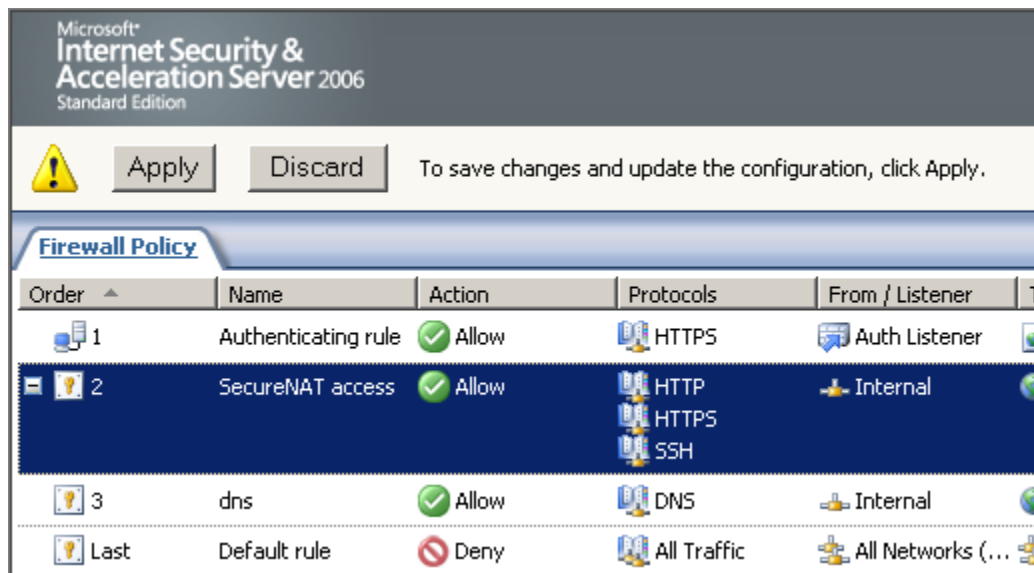
Here, we use "External" as the destination; denoting unrestricted access to the Internet:



**You must** choose “All users” since this rule needs to apply to SecureNAT traffic. But we will be using the Captivate filter to associate users with IP addresses anyway.



**Make sure** the access rule is **below** the publishing rule, or else requests for the authenticating listener will try to go out the wrong pipe:



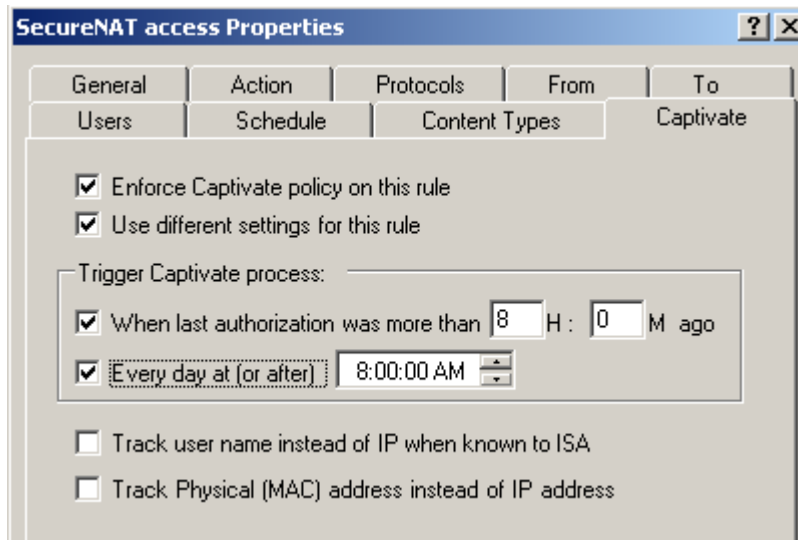
Apply your changes at this stage.

### ***Captivate settings for the access rule***

Go into the Properties of the access rule just created, and select the Captivate tab. Fill

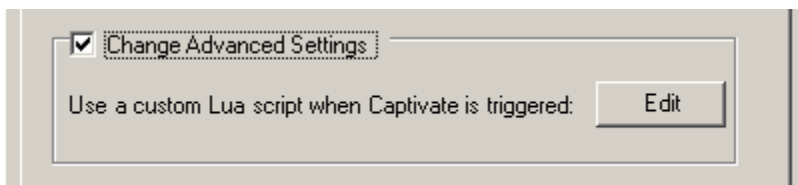


it out as below. You can substitute different trigger intervals than the ones shown. These times determine how often to show the authentication screen for a given IP address.

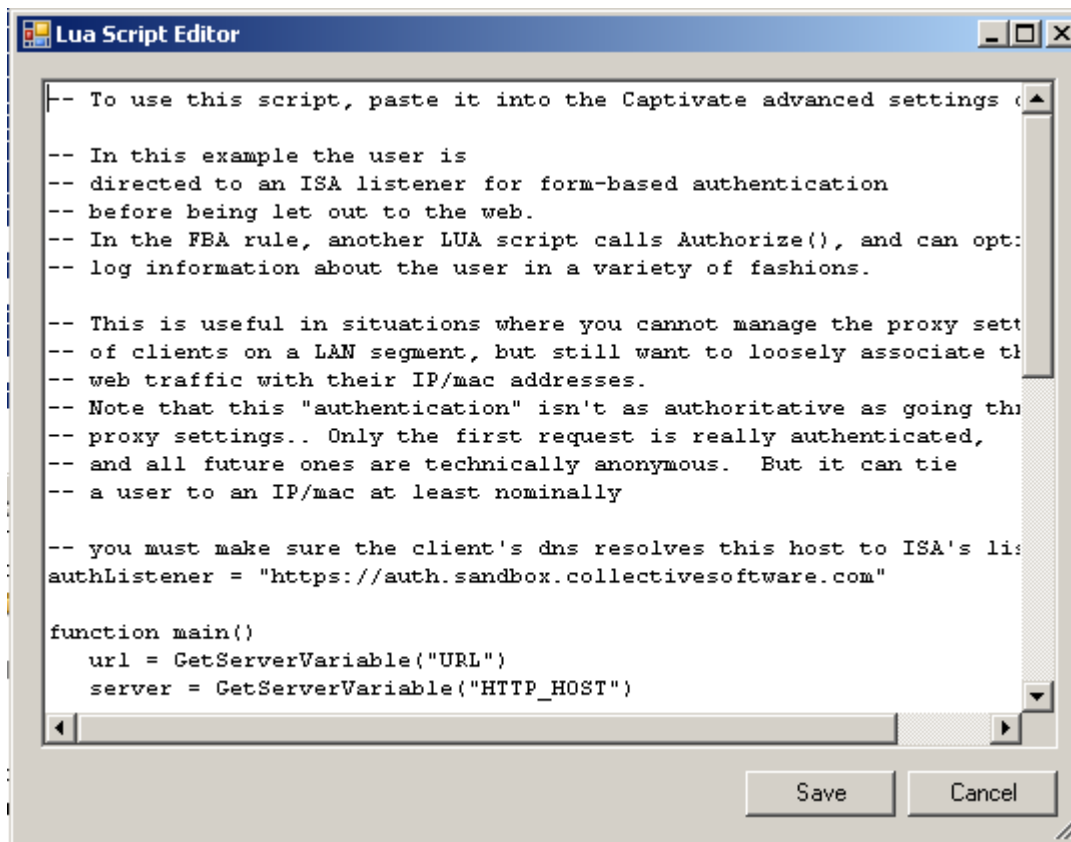


Next, we need to set up the behavior of the filter for this rule. When the Captivate process is triggered, this rule needs to save the user's requested URL and forward the browser to our authenticating listener. There is an included lua script to do this: [ISAFolder]\Collective Software\Captivate\lua\examples\Authenticate.lua.

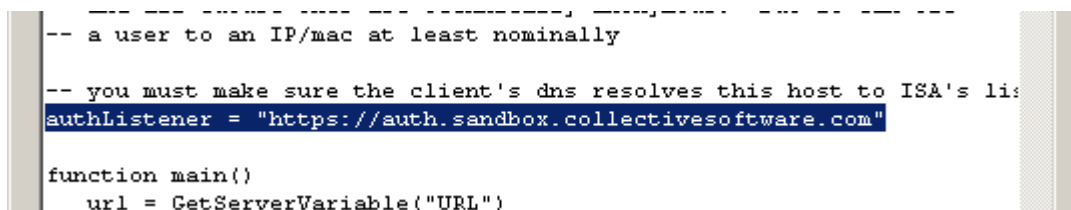
Select "Change Advanced Settings" and Edit:



and then paste the contents of Authenticate.lua into the editor window:



Scroll down to the line beginning “authListener = “ You must set this to reflect the name of the Authentication publishing rule (and certificate). If you do not correct the default setting in the example file, the redirect process will fail to send users to the FBA!



Save and *Apply* these changes.

### ***Authentication events in the web proxy log***

You don't need to set up an ODBC database just to track authentications; the web proxy log will show a row like this each time there has been a successful authentication event. Look for rows matching the name of your authenticating rule, and with a username other than “anonymous”:

Action	Rule	Client IP	Client Username	Source Network	Destination
we...		192.168.4.64	anonymous		
ate...	SecureNAT access	192.168.4.64		Internal	Ext
ate...		192.168.1.150		Local Host	Ext
ed ...	Authenticating rule	192.168.4.64	isa2006se_wg\administrator	Internal	
we...	SecureNAT access	192.168.4.64	anonymous	Internal	Ext
ed ...		192.168.1.150		Local Host	Ext

The “Action” will show “Failed” simply because the Captivate filter handled the rule and did a redirect. This does *not* mean that the authentication failed! If you see a client username other than “anonymous”, that is a successful authentication event. The client IP and Log time shows where/when the user is authenticating. These items are the only way to correlate other traffic with users, because there is no way to support “true” sessions with SecureNAT.

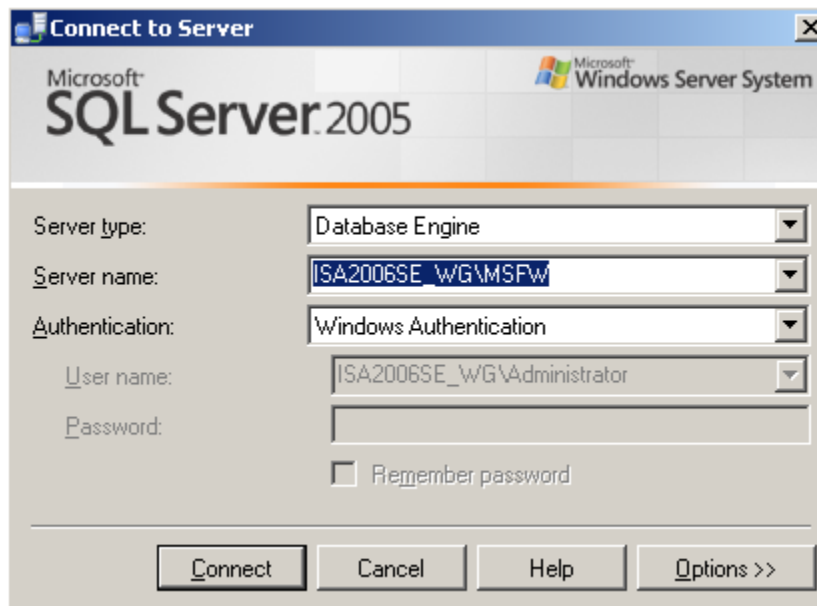
If you have ISA configured to save web proxy logs, then these authentication rows will be saved to that log as well.

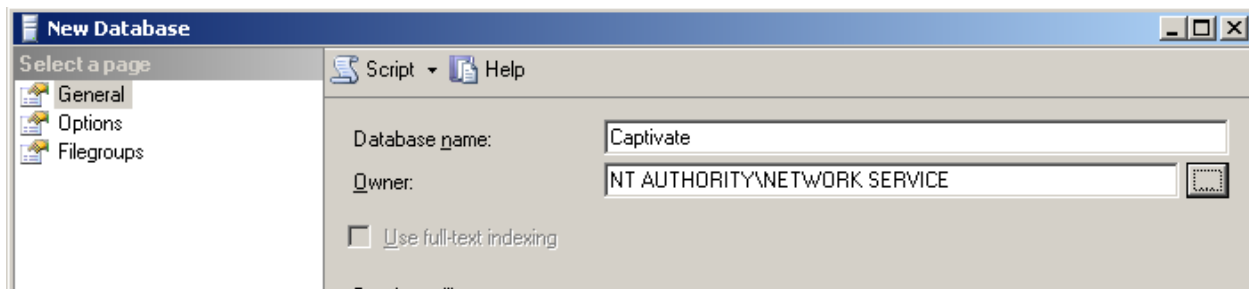
### ***Creating a Captivate database***

Depending on your requirements, it may be beneficial to create a separate table to store each authentication event.

In this example we will be logging the authentication events to the same SQL instance that is used for proxy and firewall logging. Make a separate database name to use, called Captivate, and make sure that the “Network Service” account has the dbowner role. This can be done through the SQL enterprise manager as summarized below.

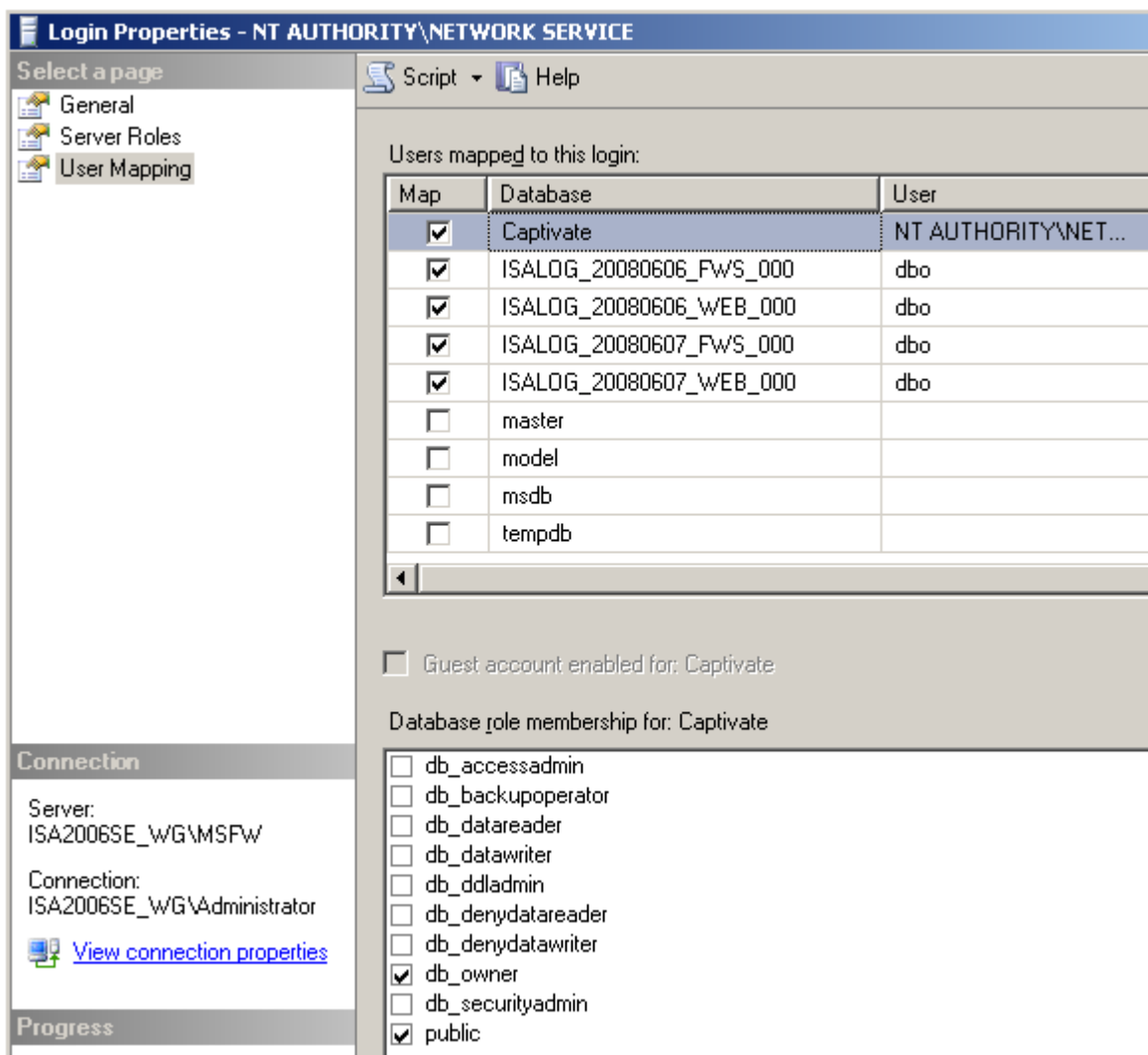
Connect to the MSFW instance on the ISA server:





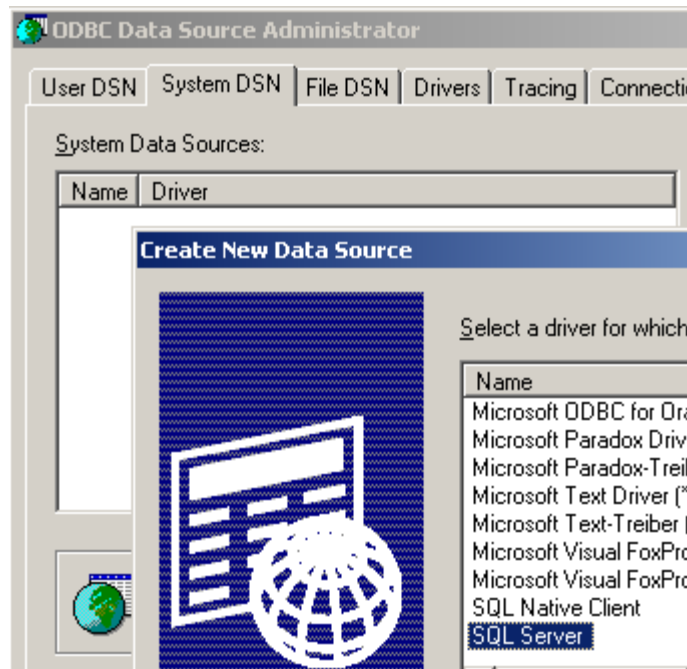
(You may receive an error about the Network Service user upon creation; this is OK, the database is still created).

Set up the user mapping so Network Service gets dbowner rights on the new db:

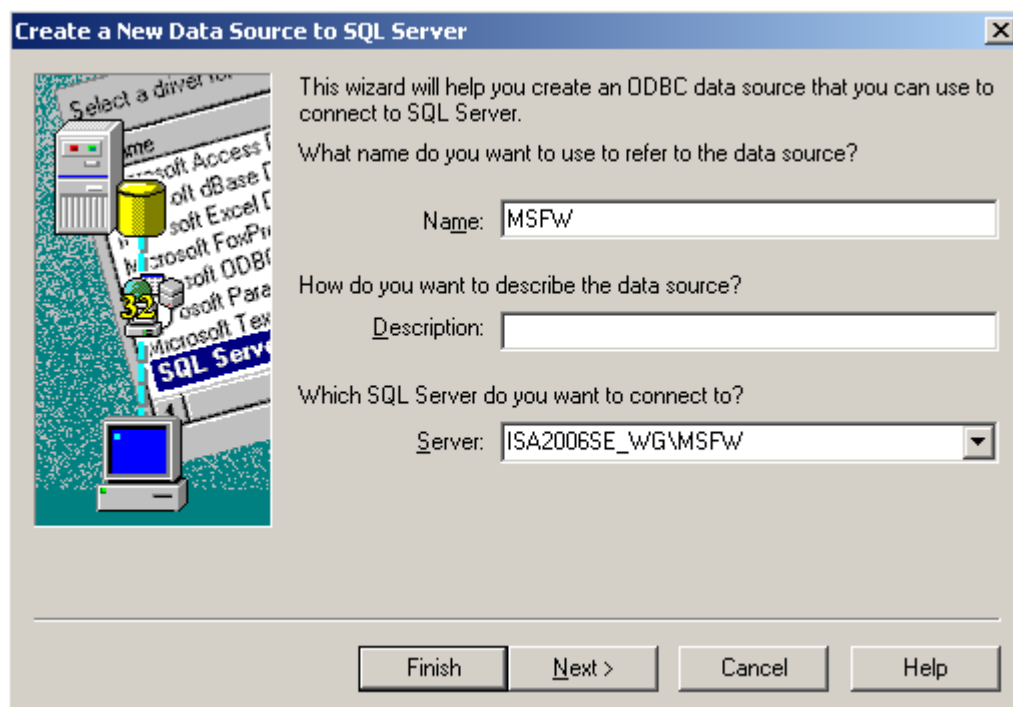


## Setting up the ODBC source

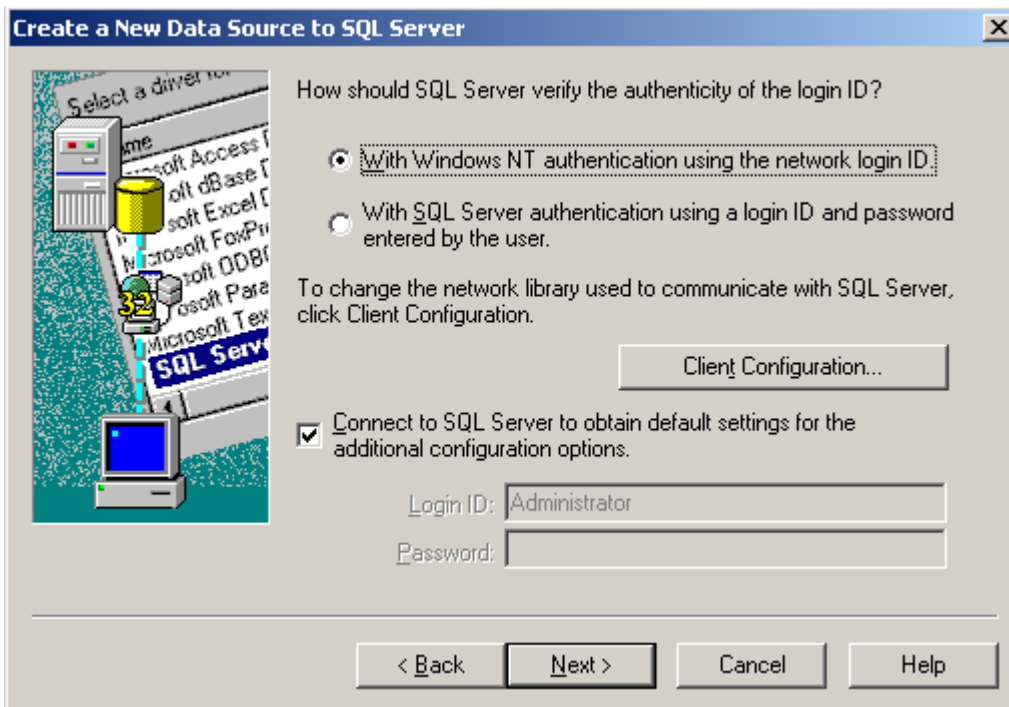
Select "Data Sources (ODBC) from Windows control panel, and add a System DSN of type "SQL Server".



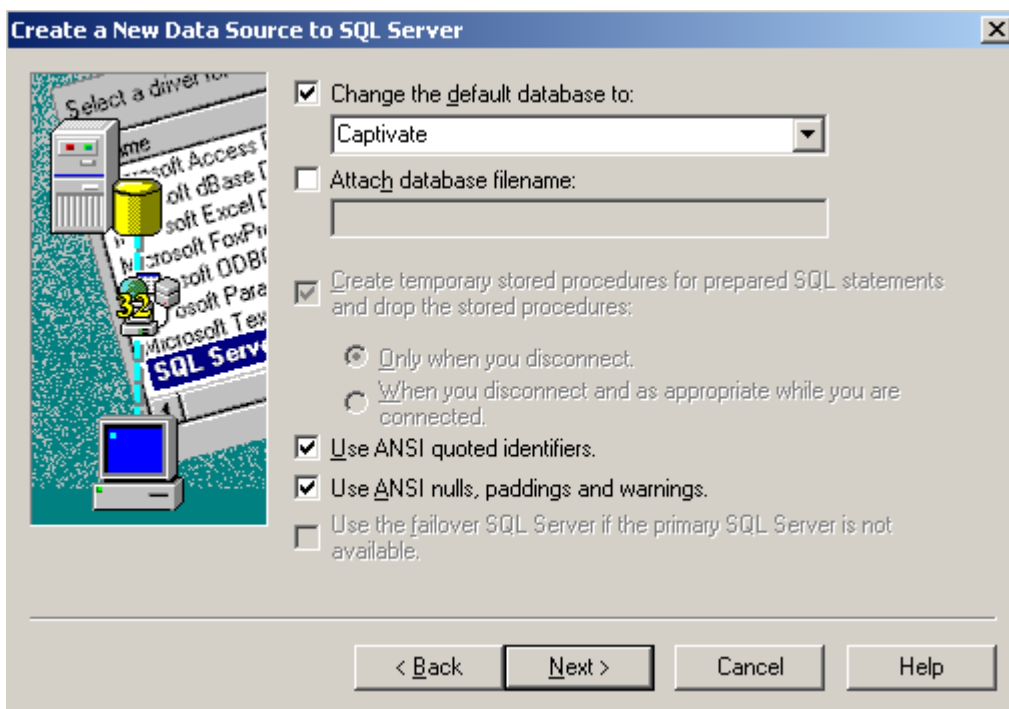
Set the name to “MSFW” and the server to the name of the ISA server followed by MSFW:



Use Windows auth:



and set Captivate to be the default database:



Configure other items to meet your needs, or leave the defaults. Make sure the data connection test is successful.

When authentication events occur, a row will be created in the Authorizations table in the Captivate database:

	id	auth_time	user	ip.
▶	1	6/6/2008 8:24:1...	ISA2006SE_WG\...	192.168.1.1
*	<i>NULL</i>	<i>NULL</i>	<i>NULL</i>	<i>NULL</i>

Each row contains the time, username, IP address (in readable and ISA-like numeric formats), MAC address if known, and the URL that the user was requesting at the time the authentication happened. Note that there will only be one row per authentication event, *not* one row per web request. There is no way to do true session association with SecureNAT clients.

### **Client checklist**

- Client machines must use the IP of ISA as the default gateway for SecureNAT connectivity to work.
- Client machines' DNS must be able to resolve the name of your Authentication listener (in our example “auth.sandbox.collectivesoftware.com”) to the IP of ISA facing the clients' network.
- Client browsers should not have proxy settings configured.
- Client machines should trust the certificate used in the authentication listener.
- Recall that once you have authenticated, that IP address will be treated as authenticated until the Captivate trigger interval has elapsed. There is no way to do “true” sessions for SecureNAT clients; we can only attach usernames to the IP addresses periodically for reporting purposes.

## Filter licensing

To view your evaluation period or enter a key, go to Add-ins, Web Filters, and select Captivate properties, and select the License tab.

The License tab is used to check how long remains in the evaluation period, and to activate a permanent license.

To be eligible for a license key, you need to purchase license(s). You can do this on our [web store](#) or by [contacting us](#).

Once you have available license(s) you can request a key for your array (or single server) at our [licensing page](#). When requesting a license key, you will need to tell us the name of the ISA array, which is indicated on this dialog. The exact name is important, because it will be used to validate the key.

The license key is sensitive to the number of servers in the array. For example if you begin with only 2 servers in the array but plan to have 4, you can purchase 4 licenses and request a license key for a 4-server array. Then as you bring future servers online, they will be licensed automatically (you still need to [install the certificates](#) though.)

**Warning:** if you install more servers than you have licensed then the license key will be seen as invalid, and the servers will begin to operate in [demo/lab mode](#). So if you need to add more servers to a live array then you should acquire and apply your new license key in advance, so this behavior does not take place.

### ***Demo/Lab mode***

When the evaluation period expires (after 30 days) or when an invalid license key is used, the filter runs in demo/lab mode. In this mode the filter will work normally for a period of 2 hours from the starting of the Firewall Service, and then stop working after that time. This mode is meant to be useful for test labs where you don't wish to purchase licenses but still want to be able to run meaningful test setups. After 2 hours, you can restart the firewall service and the lab timer will reset again.

### ***Troubleshooting***

The first place to look if something seems to be working incorrectly is the ISA alerts tab in the Monitoring section. Often this will directly indicate the cause of the problem. This information will also be required in almost all cases if you need support.

## Support for Captivate for ISA Server

Collective is proud to offer support for Captivate for ISA Server, whether you need help getting a configuration working, find a bug, or just have a feature question.

Support is available from our web site at <http://www.collectivesoftware.com/Support/>

- *Knowledge Base:* When our staff answers questions that will apply to the whole community, they will often create a permanent KB item to disseminate this knowledge. There is a Search feature here; you can also easily browse by topic.



To get fast answers to FAQs (frequently asked questions) the knowledge base is the best place to start.

- *Support ticket:* We are always happy to help you get set up and working. If you have questions or need assistance understanding/configuring/testing a Collective product, you can get in touch with our support staff quickly and easily. For the most up-to-date information, please see our Support page.