Division of IT

Shibboleth SP Simple Installation Guide For Windows and IIS

University of Missouri

October 2012

1. Background

1.1.What is a Service Provider?

To put it simply, a service provider is the website you are trying to secure. Users authenticate to your site and you provide a service to them or send them away if they are not authorized to use your site.

For Windows installations, the service provider has a windows service and an ISAPI module that loads into IIS. Like Linux, on Windows these two components work together with the Shibboleth IdP to authenticate users and provide information to your IIS application.

1.2. How does my application integrate with the Shibboleth SP software?

The Shibboleth SP service and IIS ISAPI modules provide your application with one or more environment variables that you may use in your code to determine which users are visiting your site. If you've written your application to use REMOTE_USER or a similar variable then you can probably use Shibboleth. There are issues when using REMOTE_USER in IIS, so the Shibboleth authors recommend picking a different variable name for a username, that name can be configured during the SP software setup.

Commercial applications that can use an environment variable for a username can often also be configured to use Shibboleth. At the University of Missouri, Shibboleth has been used to secure commercial applications like Coldfusion, Oracle Apex, Blackboard, and PeopleSoft's portal. The key to making this work is that the application must turn off its default authentication system and rely on authentication services configured in IIS.

1.3.Can Shibboleth tell me if a visitor is enrolled in a specific course or whether they are enrolled as a full or part-time student?

Shibboleth is best used as an "authentication" service. That is to say it tells your application who a person is (his or her username) and to a limited extent what roles within the institution the person may have, e.g. "staff", "student", etc. Shibboleth performs all lookups for all users that login. Having application specific authorization criteria will make the login process slower and less reliable for all applications. If your application needs more explicit information about a visitor, it is best if the application takes the username information and queries other data sources to determine whether that user is authorized to access a particular resource.

1.4.What if my application doesn't run under IIS?

Most windows web applications use IIS as the engine that handles requests from users' browsers. Shibboleth doesn't run on anything but Apache or IIS. It may be possible to run an SP on Windows servers using Apache, but this document focuses on IIS integration. For Apache integration, please consult the Linux install guide and the Shibboelth documentation site. That said, it is also possible to use IIS to protect web applications that run in a Tomcat, WebSphere, JBoss or other application environment with Shibboleth. You simply have to use an ISAPI module to connect IIS to Tomcat or other application server. This will put IIS/Apache "out in front" of your application server to handle security, then the IIS/Apache will pass user credentials through to your application. See Appendix A for further discussion

of using IIS/Shibboleth as a "reverse proxy." This method of securing applications is very common, but requires a thorough understanding of the underlying application.

2. Installing a Simple Service Provider

This section describes the process of installing a simple IIS 7 backed Service Provider (SP) that communicates with the University of Missouri System Shibboleth Identity Provider (IdP) from the initial request to succesful integration in the production environment.

For a comprehensive installation guide please see the Native SP Windows install pages on the Shibboleth Wiki. https://wiki.shibboleth.net/confluence/display/SHIB2/NativeSPWindowsInstall

Prerequisites for Installation:

- A basic understanding of web applications in general as well as your application and in particular how your application authenticates users.
- For Windws SPs: A remote desktop client, the ability to navigate directories and use a text editor for windows.
- A way to install the SP software on your host either with root/administrator access or by having someone who can install the software for you.
- Permissions to modify the shibboleth configuration files, your Apache configuration files or IIS configuration AND the ability to start and stop the shibboleth service as well as IIS.
- You **must** have already configured your website with an SSL certificate and at the very least set up a sample application or directory that you can use to test that Shibboleth is working. SSL certificates that are recognized by nearly every browser are available free of charge for University domains from your campus IT organization or the Division of IT SSL certificates page: https://doit.missouri.edu/security/certificates/

2.1.Service Provider Request Process

Any University application owner who wishes to integrate with the University of Missouri Shibboleth Identity Provider should complete the request form which can be found on the IT Security Shibboleth page (<u>http://doit.missouri.edu/security/shibboleth/</u>). The completed form will be automatically submitted to the **UM DoIT Shib Support** distribution list for processing.

2.2. Prepare Windows by Installing IIS and other Pre-requisite Applications

Chose your OS/IIS/Shibboleth architecture. Your **entire** IIS, Shibboleth and application (.Net etc) software environment must be the same hardware architecture (32-bit or 64-bit). You **cannot** mix architectures on Windows. Even though 32-bit applications will usually run on a 64-bit OS, this is NOT the case with Shibboleth. It **will** fail if you run in a mixed architecture environment; often without anything to indicate why. Before downloading Shibboleth, verify the architecture you are using for your

application. If you are using 32-bit, it is best to install a 32-bit version of windows just to avoid any potential mixing of software architecture components.

Note: If you are adding Shibboelth SP support to a server that already has several IIS applications installed and running, the Shibboleth installer may have trouble configuring the main IIS applcation container and you may be required to configure Shibboleth for each of your application sites individually. It is best (but not required) to install Shibboleth in a clean IIS environment.

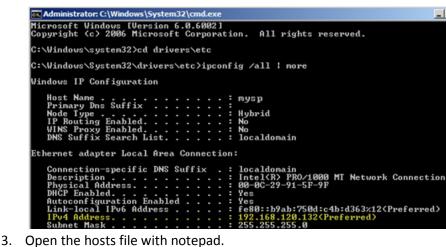
2.2.1. Set Your Machine's HOSTNAME in the hosts File

Setting the server's hostname in the c:\windows\system32\drivers\etc\hosts file will make the install and configuration changes go smoother.

1. Open a command prompt window as administrator by typing "cmd" in the search box and in the resulting list, right-click cmd and choose "Run as administrator".

Progra	ms	
os. cmd	Open	
	Run as administrator Scan with ClamWin Free Antivirus	Administrator
	Edit with Notepad + + Pin to Start Menu	Documents
	Add to Quick Launch Restore previous versions	Computer
	Send To	Network
	Cut Copy	Control Panel
	Delete	Administrative Tools
	Open file location Properties	Help and Support
		Run
1	h Everywhere	
Search	h the Internet	
cmd		0 🔒 🕨

2. Change directories to c:\windows\system32\drivers\etc and run the following command in the command prompt window to find your machine's IP address "ipconfig /all | more". Look for your machine's IP address (highlighted in the picture) and remember it.



- C:\Windows\System32\drivers\etc>notepad hosts
- 4. In notepad leave the entries for 127.0.0.1 and ::1 and add a new line with your machine's IP address and its fully qualified hostname (hostname.domain.edu) followed by a space and then just the host name.

127.0.0.1 localhost ::1 localhost 192.168.120.132 mysp.umsystem.edu mysp

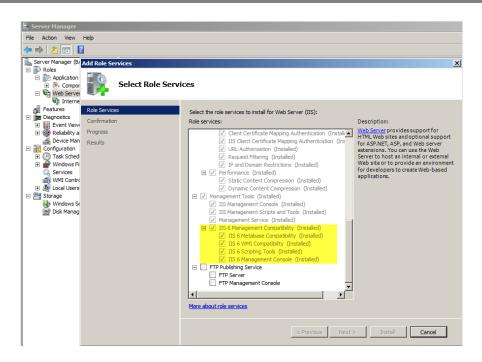
5. Save your hosts file and close notepad and the command window. If you recieve a permissions error on the file, then you most likely did not open the command window with "Run as administrator" permissions.

Warning: Do NOT use the underscore character in your machine name. The underscore is NOT a valid character in standard DNS host or domain names. Several shibboleth and IIS components use the computer's name to communicate with each other. A machine name with an underscore may break this communication.

2.2.2. Verify IIS 6 Roles and ISAPI Filters are Installed

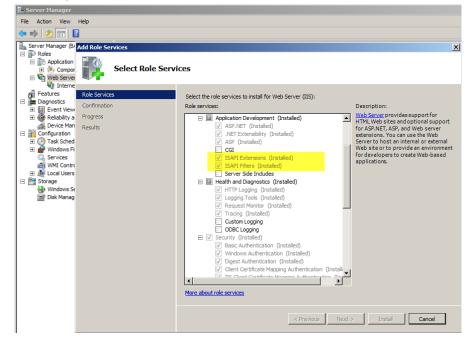
The Shibboleth SP software was written for IIS 6, but is compatible with IIS 7 which is the default install on Windows 2008 R2. To simplify installation and configuration you **must** install the "IIS 6 Management Compatibility" on your Windows server **before** installing Shibboleth.

- 1. Click Start, click Administrative Tools, click Server Manager
- 2. In the left pane, expand the Roles tree and right-click Web Server (IIS), then seelct "Add Role Services".
- In the Role Services window scroll to the bottom and verify that IIS 6 Management Compatibility" and all of its sub-components are installed. If they are not installed, check the boxes and install them.



Verify that support for ISAPI filters and extensions are installed with IIS.

- 1. Click Start, click Administrative Tools, click Server Manager
- 2. In the left pane, expand the Roles tree and right-click Web Server (IIS), then seelct "Add Role Services".
- 3. Scroll to the "Application Development" section of the Role Services pane.
- 4. Verify that "ISAPI Extensions" and "ISAPI Filters" are installed or install them if necessary.



6

2.3. Download Shibboleth SP Software for your OS

Shibboleth software is available in pre-built installers for Windows. We do not recommend trying to build Shibboleth from source code on Windows. You must install the version appropriate to your operating system and choose the correct architecture (32-bit or 64-bit). Use at least SP version 2.4.3 or higher. Earlier versions have a serious security hole in one of the accompanying software libraries.

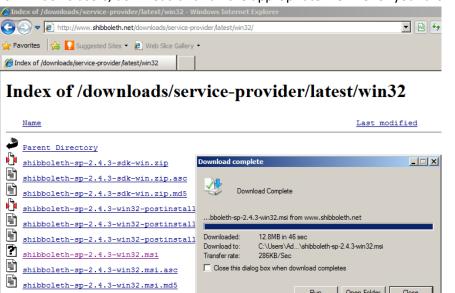
Shibboleth Service Provider (SP) software can be found at the link below:

http://shibboleth.net/products/download.html

2.4.Install Shibboleth SP Software

Note: Installing the SP software requires a reboot of the OS.

1. For windows users, download and run the appropriate MSI file for your architecture.



Open Folder

Run

Close

2. Run the Installer. It is not signed so you will see a security warning.



Proceed through the shibboleth installation. Accept the license agreement.

🛃 Shibboleth 2.4.3 Setup		🖓 Shibboleth 2.4.3 Setup	
INTERNET. MIDDLEWARE	Welcome to the Shibboleth 2.4.3 Installation Wizard	License Agreement You must agree with the license agreement below to proceed.	
MIDDLEWARE INITIATIVE	It is strongly recommended that you exit all Windows programs before running this setup program. Click Cancel to quit the setup program, then close any programs you have running. Click Next to continue the installation.	Apache License Version 2.0, January 2004 http://www.apache.org/licenses/ TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION 1. Definitions	
	Shibboleth.	1. Deminuous. "License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document. "Licensor' shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.	•
	< Back Next > Cancel	I accept the license agreement I do not accept the license agreement Reset Above: Above:	Cancel

4. Install the SP software into the default location of C:\opt\shibboleth-sp\. Attempting to install anywhere else may require changes to the configuration files.

🚏 Shibboleth 2.4.3 Setup	_ 🗆 🗙
Destination Folder Select a folder where the application will be installed.	S
The files for Shibboleth 2.4.3 will be installed in the following folder.	
To install into a different folder, click the Browse button, and select another folder.	
You can choose not to install Shibboleth 2.4.3 by clicking Cancel to exit the installe Destination Folder C:\opt\shibboleth.sp\ Browse	er.
DO NOT use a path containing any spaces. This is a limitation of the Shibboleth configuration file and will result in a broken installation.	ne
Kent Next Next Next Next Next Next Next Nex	Cancel

5. Let Shibboelth install its service on the default port 1600. The IIS ISAP module and the Shibboleth SP service will communicate with each other on this network port. Use Windows' firewall features to block access to this port from other computers. The port is not normally exposed to the network if windows firewall is active; verify that this is the case.

Shibboleth 2.4.3 Set	up	
Shibd Service Specify the port for the	Shibd daemon.	X
NOTE: This is NOT that is NOT in use I	P port number on which the shibd daemon should the web server port and MUST be a valid Windo w any other process! ath.xml already exists in the installation path, it will i umber.	ws port number
	Install Shibd daemon	
	< Back Nex	t> Cancel

6. Allow the installer to register the default ".sso" file extension handler in IIS.

🙀 Shibboleth 2.4.	3 Setup	_ 🗆 X
Install ISAPI Filter Install the ISAPI		S
	I the ISAPI filter for IIS, register the specified file extension and create a Extension entry (IIS 6 and later).	
	Install ISAPI filter and configure IIS Shibboleth File Extension: sso	
	<back next=""> C</back>	ancel

7. The Installer will show a progress bar while it copies files and sets up the services and environment. Click "Finish" and be prepared to reboot your server when prompted.

🙀 Shibboleth 2.4.3 Setup		×				
INTERNET. MIDDLEWARE INITIATIVE	Shibboleth 2.4.3 has been successfully installed.					
	Click the Finish button to exit this installation.			ler Information		X
	Shibboleth.		2	You must restart your system to Shibboleth 2.4.3 to take ef if you plan to restart manually	for the configuration changes made frect. Click Yes to restart now or No later.	
	< Back Einish Cancel			Yes	No	

The installation software will automatically create several default configuration files in c:\opt\shibboleth-sp\ and generate a private key and certificate that is used to encrypt/sign data sent to and received from the Shibboleth IdP. We strongly encourage you to use the default key and certificate created by the installer. This certificate is not used by IIS, and you do not need to use the same key/cert for Shibboleth as you are using to protect your web site. Using the same key and certificate for your web site and the Shibboleth service will NOT improve your site security and will only make administering your shibboleth configuration more difficult. The default "self signed" Shibboleth certificate will expire in 10 years.

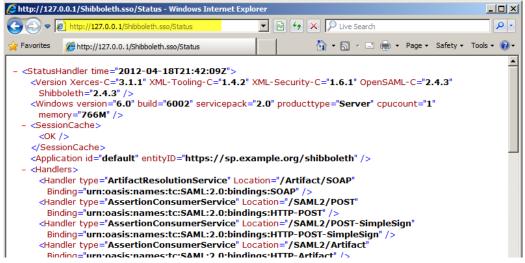
2.5.Configure Service Provider Metadata

Metadata is the data used to configure and describe your Shibboleth SP, and there are seemingly an infinite number of configuration options. However, for most installations only two or three files need to be changed to get Shibboleth up and running.

2.5.1. Verify that Shibboleth SP Software is Installed Correctly

After your server has rebooted, it is important to verify that Shibboleth installed correctly and starts automatically. Even though the SP software is not yet configured, it is still capable of producing a status report. Viewing the report means that the SP service is running and that the ISAPI filter was correctly installed in IIS.

- 1. Open an InternetExplorer window and open the following URL:
 - http://127.0.0.1/Shibboleth.sso/Status. You should see a bunch of XML returned.



2. If you do not see XML then it is possible IIS isn't running or that access restrictions are in place to prevent the status message from appearing. Open a command window as administrator, and restart IIS by running the iisreset command.

```
C:\Users\Administrator>iisreset
Attempting stop...
Internet services successfully stopped
Attempting start...
Internet services successfully restarted
```

Change to the shibboleth install directory: \opt\shibboleth-sp\etc\shibboleth and edit the shibboleth2.xml file with notepad.

C:\Users\Administrator>cd \opt\shibboleth-sp\etc\shibboleth\ C:\opt\shibboleth-sp\etc\shibboleth>notepad shibboleth2.xml

Inside notepad scroll to the lower half of the shibboleth2.xml file and edit the section that starts with '<Handler type="Status"'. Modify the acl= attribute to include "::1" which is the IP version 6 "loopback" address. Also add any other IP addresses you wish to have access to the status page. Warning: it is NOT possible to add entire networks or subnets; only individual IP addresses will work. Addresses are separated by spaces.

<!-- Extension service that generates "approximate" metadata based on SP configuration. --> <Handler type="MetadataGenerator" Location="/Metadata" signing="false"/>

<!-- Status reporting service. --> <Handler type="Status" Location="/Status" acl="127.0.0.1 ::1 192.168.120.132 192.168.120.8"/>

<!-- Session diagnostic service. --> <Handler type="Session" Location="/Session" showAttributeValues="false"/>

5. Restart the Shibboleth SP service by opening the services MMC snap-in, finding the Shibboleth SP service in the list and stoping and restarting it.

C:\opt\shibbo	leth-sp\etc\sh	ibboleth	>mmc services.m	sc				
🖏 Services							_	
File Action View	Help							
	à 📑 🛐 🖬 🕨 🗉							
Services (Local)	🔅 Services (Local)							
	Shibboleth 2 Daemon	(Default)	Name 🔺	Description	Status	Startup Type	Log On As	
			Remote Procedure	Manages t		Manual	Network S	
	Stop the service		🧟 Remote Registry	Enables re	Started	Automatic	Local Service	
	Restart the service		🌼 Resultant Set of Po	Provides a		Manual	Local System	
	5		🧟 Routing and Remot	Offers rout		Disabled	Local System	
			🌼 Secondary Logon	Enables st	Started	Automatic	Local System	
			🌼 Secure Socket Tun	Provides s	Started	Manual	Local Service	
			🔍 Security Accounts	The startu	Started	Automatic	Local System	
			🔍 Server	Supports fil	Started	Automatic	Local System	
			🔍 Shell Hardware Det	Provides n	Started	Automatic	Local System	_
			👫 Shibboleth 2 Daem		Started	Automatic	Local System	
			SL UI Notification S	Provides S	Started	Manual	Local Service	-
			🔍 Smart Card	Manages a		Manual	Local Service	
			🔍 Smart Card Remov	Allows the		Manual	Local System	
			SNMP Trap	Receives tr		Manual	Local Service	
			Software Licensing	Enables th	Started	Automatic	Network S	

- 6. Check the status page again.
- 7. If you still do not see a status page, then most likely the installation is corrupt. Uninstall and reinstall the shibboleth-sp software. Rebooting after each step.

2.6.1. Modifying shibboleth2.xml

Configuring the Shibboleth SP involves editing XML files in c:\opt\shibboleth-sp\etc\shibboleth. It is strongly recommended that you make a backup copy of each file before editing and check each modification by running the "shibd -check" command (in a command prompt window). When running "shibd -check" the result should reveal an error message or "Overall configuration is loadable".

The shibboleth2.xlm file contains the basic Shibboleth SP configuration, on windows there is a special <ISAPI> section near the top of the file. The ISAPI section does not appear in non-windows installs.

- Set the safeHeaderNames as you see fit. When set to true it will remove all dashes and underscores in header names before they are sent to applications. For example SAM_ACCOUNT_NAME will become SAMACCOUNTNAME. This overrides any attributeID names set in attribute-map.xml. Note that if IIS prepends a header with HTTP, then the header will be HTTP_SAMACCOUNTNAME not HTTPSAMACCOUNTNAME.
- 2. Map one or more IIS "sites" to shibboleth using the <Site> tag. Set the id attribute to match your IIS site number and the name attribute to match the site host name. Optionally set the scheme attribute to "https" and specify a port for (typically 443) the scheme. Site IDs and the host/port they run on are visible in the IIS configuration manager as shown below:

Hinternet Information Servi	ces (IIS) Manager			
BARFYSP	▶ Sites ▶			
File View Help				
Connections	Sites			
Start Page ⊡ • G BARFYSP (barfysp \Admir	Filter:	- 🔐 Go	- Show All	₌
Application Pools	Name 🔺	ID	Status	Binding
⊞ <mark>ii</mark> Sites	😌 Default Web Site	1	Started (http)	*:80 (http)

3. Save the shibboleth2.xml file and re-start and recheck the status.

The c:\opt\shibboleth-sp\etc\shibboleth\shibboleht2.xml file contains a unique identifier for your SP, sets session timeout information and tells the SP software about the IdP (where people sign-in). Open your shibboleth2.xml file with your favorite text editor and make the following changes:

1) Search for the <ApplicationDefaults tag and set your entityID to the hostname URL for your website followed by a "/" and the word "shibboleth". This entityID is completely arbitrary and does NOT point any browser or user to any specific location on your site. You could just as easily name your entityID something like "super.waycool.funzone". However, it is very important to use your machine's URLs because URLs represent unique values on the internet, and they make it easy for administrators to identify a particular SP configuration. If you plan on using a load balancer in front of your server, use the server name your users would see in their browser, not the individual host names of the multiple servers behind the load balancer. From the IdPs perspective, multiple SPs behind a load balancer should have an identical configuration. Here is a valid example :</p>

entityID="https://myiissite.umsystem.edu/shibboleth"

2) Unlike Linux, the REMOTE_USER variable is not populated correctly by Shibboleth and IIS. For completeness you may choose to configure the variable anyway. In the same ApplicationDefaults tag as step 1 above add the value "samAccountName" to the REMOTE_USER attribute. For example:

REMOTE_USER="samAccountName eppn persistent-id targeted-id"

This will attempt to set the REMOTE_USER web server environment variable to the first available attribute from the list: samaccountname, eppn, persistent-id, and finally targeted-id. It will most likely fail on IIS, but it should set the HTTP_REMOTEUSER variable instead.

3) Search for the <Sessions tag and make it look like the code below. You can modify the numbers for lifetime and timeout, they are recorded as seconds:

```
<Sessions lifetime="32400" timeout="32400" checkAddress="false"
consistentAddress="true"
```

```
relayState="ss:mem" handlerSSL="true"
postData="ss:mem" postTemplate="postTemplate.html"
cookieProps="; path=/; secure">
```

The settings shown above will set your Shibboleth session lifetime to 9 hours (the maximum allowed by the University's IdP). It also sets the timeout, which is the maximum time allowed between visits to your site for a session to remain active, to 9 hours. It also sets web form POST operations to save their data in memory cache rather than discarded when a user requires authentication. Finally, it sets the Shibboleth cookie properties so that the cookie applies to the entire site. It's only sent when the browser connects with https (SSL encryption), and tells the browser to delete the Shibboleth cookie when the browser is closed. To use the "secure cookie" settings, your site MUST use HTTPS (SSL) not HTTP. You should have already configured your site to use SSL in the pre-requisite section at the beginning of this document.

4) Now configure the SP to point to the University's IdP. Search for the first <SSO tag and set it as follows:

```
<SSO entityID="https://shib-idp.umsystem.edu/idp/shibboleth"
discoveryProtocol="SAMLDS"
discoveryURL="https://shib-idp.umsystem.edu/DS/WAYF">
SAML2 SAML1
</SSO>
```

The settings above point your SP to the University's only IdP for single-sign-on and "discovery". We don't actually use discovery, but if we did this is where your SP would send users.

5) As mentioned above in section 2.5.1 search for <Handler type="Status" and add your desktop workstation's IP address to the ACL list. This will allow you to view the Shibboleth "status" URL from your desktop. Add other IP addresses as needed (if you have monitoring software that can monitor a site URL, etc). Do not attempt to add whole subnets or open the status to the world. Your status handler line should look something like this:

```
<Handler type="Status" Location="/Status"
acl="127.0.0.1 ::1 128.206.92.200"/>
```

6) Search for <Handler type="Session" and change the showAttributeValues to true. Later you can change this back to false, but for setup and debugging, it is really helpful to have Shibboleth be able to display data about logged in users. Your session handler should look like this:</p>

```
<Handler type="Session" Location="/Session"
showAttributeValues="true"/>
```

7) Search for <Errors supportContact and adjust the support contact email address and logos to fit your site "look and feel". If you don't, your users will see the standard Shibboelth "griffen" when a Shibboleth error occurs. Adjust your settings to point at your version of these files, perhaps something like this:

```
<Errors supportContact="oscarthegrouch@missouri.edu"
    logoLocation="/images/logo.jpg"
    styleSheet="/css/main.css"/>
```

8) Search for <MetadataProvider and configure the IdP metadata as follows:

```
<MetadataProvider type="XML"
uri="https://shib-idp.umsystem.edu/idp/profile/Metadata/SAML"
backingFilePath="shib-idp.umsystem.edu.metadata.xml"
reloadInterval="7200"/>
```

The settings above tell your SP to retrieve the IdP metadata dynamically every 2 hours from the IdP itself and to save it in a temporary file called shib-idp.umsystem.edu.metadata.xml. This is the **best way** to keep your IdP metadata file up to date. It is also possible to retrieve the file manually and place a static copy of the IdP metadata on your server, but this is discouraged.

- 9) You may also change the default error page templates. They are located in the same direcory as shibboleth2.xml (c:\opt\shibboleth-sp\etc\shibboleth*.html).
- 10) Save your changes.

2.6.2. Configure the attribute-map.xml file

The attribute-map.xml file sets up the user attributes that your site accepts from the IdP. For the purposes of this document, we'll only request the user's logon userID or samAccountName as it is called in ActiveDirectory. Make the changes shown below to your attribute-map.xml file:

1) At the top of the file insert the following:

```
<Attribute name="urn:mace:umsystem.edu:attribute-def:samAccountName"
id="samAccountName">
```

```
<AttributeDecoder xsi:type="StringAttributeDecoder"</pre>
```

```
caseSensitive="false"/>
```

```
</Attribute>
```

This will allow the SP to recieve the samAccountName attribute if it is sent by the IdP. Be very careful with capitalization; it can sometimes bite you even when caseSensitive = false.

2) You may uncomment other attributes, but attempting to process attributes you don't use will cause your SP software to run slower. Do not configure any attributes you don't need. Adding attributes here **DOES NOT** mean that you will automatically receive them from the IdP, nor does it tell the IdP that you wish to receive them. The IdP configuration is solely responsible for the attributes sent to your SP and you **must** tell the Shibboleth Support team which

attributes you need (do this using the request form mentioned at the beinning of this document). The attribute-map.xml file only tells your Shibboleth SP software which attributes, of those it does recieve, are to be passed on to applications.

3) Save your changes.

2.6.3. Test the SP Configuration

After the changes above, you should test your SP configuration again.

- 1. Use shibd from the command line to check your config
 - c:\opt\shibboleth-sp\sbin\shibd -check
 - You should see the words "overall configuration is loadable"
- 2. Restart your web server (iisreset) and the Shibboleth service (Services MMC)
- 3. Check the status page again <u>https://localhost/Shibboleth.sso/Status</u>

If you don't see "Overall configuration is loadable", and instead see some other error, then most likely you didn't close an XML tag or quotes in the configuration changes you made above. Shibd will start without error using the distributed copies of the xml configuration files. If it doesn't start after you've made changes, it is most likely due to one of those changes. Set a DEBUG level in the shibd.logger configuration file, correct your mistakes and test again until your configuration loads.

2.6.4. Generate your SP's Metadata File

The entire configuration up to this point has been necessary to make the Shibboleth SP deamon/service run on your server. Now information must be exchanged with the IdP so the two Shibboleth services can communicate. To do this, the IdP needs a copy of your "metadata". There are two methods to do this.

The first way is to make your metadata available on a URL that the IdP can download. The second (preferred) method is to create your own copy of the metadata file and e-mail it to shibsupport@umsystem.edu. The URL method seems convenient because you can make changes to your metadata and have it be automatically updated on the IdP. Unfortunately, if your changes cause your metadata to break then the IdP will simply not load the changed file. You may not know you've got an error, but your changes won't work either. It's best to coordinate changes with the Shib support team by sending them your metadata in email.

Either way, it is important to understand that your SP software does not use or reference your generated metadata file at all. If you save a copy of your metadata in the Shibboleth configuration folder, it is just an extra file. The SP gets its configuration from the shibboleth2.xml and attribute-map.xml files mentioned above. It does NOT use the metadata file you save to disk or send to the Shibboleth team. The number one cause of mistakes in a Shibboleth SP configuration is for someone to change their generated metadata file, but not their actual configurations in shibboleth2.xml or attribute-map.xml.

To generate your SP's default metadata file (based on the configuration changes you've already made), do the following:

- 1. Start, or restart, the Shibboleth SP service.
- 2. After restarting the shibd daemon, download your metadata from the server's web site. You may use the browser on your SP. NOTE: it is important that you reference the metadata URL using the fully qualified domain name for your site, not merely the host name or localhost or 127.0.0.1. Your basic (auto generated) metadata file is available at your site's /Shibboleth.sso/Metadata URL. For example:

https://myiissite.umsystem.edu/Shibboleth.sso/Metadata

Using your host name will help you discover any IIS SSL certificate and site/hostname issues before you get too much farther into the configuration. Also, in deployments where you use multiple hostnames the contents of the metadata may actually change depending on the host name you use to connect. You will be hand-editing the file anyway so this isn't critical, but it may reduce editing.

Certificat	e Error: Navigation Blocked - Windows Internet Explorer		
C - C	🧟 https://		☆ ☆ 🕸
			A
8	There is a problem with this website's security certificate.		
	The security certificate presented by this website was not issued by a trusted certificate authority.		
	Security certificate problems may indicate an attempt to fool you or intercept any data you send to the server.		
	We recommend that you close this webpage and do not continue to this website.		
	🧭 Click here to close this webpage.		
	S Continue to this website (not recommended).		
	⊗ More information		
	Do you want to open or save Metadata from Contract Contract Contract Contract ?	Open Save Cancel ×	-

3. The UM standard for naming metadata files is "hostname.domain.edu-metadata.xml". Rename your file to the fully qualified hostname of your machine, or even better, use the hostname from the "entityID" section of your shibboleth2.xml file.

2.6.5. Modifying the Metadata

Unfortunately, there are some configuration options that simply cannot be set in the shibboleth2.xml file, and it is sometimes necessary to modify your metadata file directly. For example, if you want to add extra host names for your site or you want to customize the login message the user sees on the IdP then you must MANUALLY modify your metadata file. See Appendix B for an example of how to set up your metadata to handle multiple host names. Remember, modifying the metadata file **will not change** how your SP functions, all of that is controlled by the shibboleth2.xml and attribute-map.xml files.

The metadata you download from your base metadata URL will be in unix format, meaning it will have only line-feeds at the end of each line. DOS or windows formats contain a carriage return followed by a line-feed characater at the end of each line. Unfortunately this makes the default file difficult to edit

with notepad because notepad doesn't recognise just linefeeds. If you open your metadata file with notepad you'll see one big long line that wraps only occasionally. Find a decent text editor that you can use to edit both unix and dos formated text files, for example Notepad++ or Programmer's File Editor (PFE).

A common customization is to change the login logo and message users see when your site sends them to the IdP to authenticate. To do this you must manually update your metadata file before sending it to the Shib support team. To do this:

- 1. Open your metadata file with a text editor. Search for <md:EntityDescriptor. It should be the very first line in the file.
- In this EntityDescriptor tag you'll see an attribute like: xmlns:md="urn.....". Leave it there
 but move your cursor to a position after this attribute and paste the following additional
 xmlns attribute:

xmlns:mdui="urn:oasis:names:tc:SAML:metadata:ui"

- 3. You should now have two "xmlns:" attributes. All this did was add another shortcut description for the XML tags later in the file. When you add your logo and description information to the file, you will use this "mdui" shortcut, otherwise you would have to preface each tag with "urn:oasis:names:tc:SAML:metadata:ui".
- 4. Search for the <md:Extensions> tag. It is probably the third line in your metadata file. Add the following tags after the <md:Extensions> tag, but before the <DiscoveryResponse... tag.

```
<mdui:UIInfo>
	<mdui:DisplayName xml:lang="en">My Test Site</mdui:DisplayName>
	<mdui:Description xml:lang="en">A test site.</mdui:Description>
<mdui:Logo height="146"
width="148">https://myiissite.umsystem.edu/icons/logo.png</mdui:Logo>
</mdui:UIInfo>
```

Set the displayname, description and logo as appropriate for your site. Each of the tags is optional. If you specify a logoit **MUST be between 64 to 350 pixels wide and 64 to 146 pixels tall. More importantly, your logo must reside on an SSL encrypted site that is NOT protected by Shibboleth.** If you place your logo on an http (unencrypted) site, then your users' browsers will likely display a warning that the login page contains unencrypted and encrypted data. This is not acceptable and will not be allowed in production because it undermines the perceived security on the login screen. Your logo will not be placed on the Shibboleth IdP server, it is up to you to serve up this file to users.

Other common metadata changes are to add descriptions for additional host names for your web site. For example, if your site is myiissite.umsystem.edu, you may also have a version for one of the other campus domains, say myiissite.umsl.edu. For shibboleth to work correctly, both names must be listed in the metadata (the entityID remains the same). See the Apendix B for an example of modifying your metadata to support multiple host names.

2.6.6. Submit Metadata to ISAM

The metadata file must now be submitted to the ISAM Shib support team.

- If you are using the preferred method of mailing the file to the Shib support team, then transfer the file to your mail client and attach it to an email and send it to shib-support@umsystem.edu. Because Exchange/Webmail sometimes blocks attachments with .xml extensions, add a .txt extension to the metadata file just to make sure it gets through. If you are using the URL method to get your metadata to the UM Shibboleth support team, then place the metadatafile in a location that your website can serve up to the IdP servers and open any necessary firewall holes so that those servers can reach your site. The production IdP servers sit behind a load balancer and have their own distinct IP addresses that MUST be put in your firewall "allow" list. Contact the Shib support team for those addresses. Send the URL to the shib-support@umsystem.edu email address.
- The Shibboleth support team will validate your metadata file for correct syntax and then upload it to the University's Shibboleth Quality Assurance server. You will be informed when this is complete. You MUST validate your site is working against this QA server before the configuration will be migrated to production.

2.6.7. Test that the SP is Working in the Quality Assurance Environment

Your SP should now be ready to test in the QA environment. To do this you must configure a Shibboleth protected directory or location on your web server. By default, the shibboleth2.xml file will contain an entry that will protect the /secure directory on your default web site.

- 1. Navigate to your web servers document root directory. Typically, this would be c:\inetpub\wwwroot.
- 2. Create a directory named "secure".
- 3. Create a simple index.html file in the directory. Perhaps something like:

<html><body>Hello World</body></html>

See Appendix C for a sample page that will display the attributes passed by shibboleth.

4. Edit your shibboleth2.xml file to verify that the secure directory is protected. You should see a RequestMapper section something like:

```
<RequestMapper type="Native">
<RequestMap>
```

<!-- The example requires a session for documents in /secure on the containing host with http and https on the default ports. Note that the name and port in the <Host> elements MUST match Apache's ServerName and Port directives or the

```
IIS Site name in the <ISAPI> element above. -->
<Host name="myiissite.umsystem.edu">
  <Path name="secure" authType="shibboleth" requireSession="true"/>
</Host>
</RequestMap>
</RequestMapper>
```

Within the <Host></Host> tags you may add other path tags to specify more protected URL's. The example above only protects the secure directory.

- 5. If you had to add the host or change the path, then restart the shibd service and IIS.
- 6. Check your status again.
- 7. Assuming you have a Windows workstation, you must now edit your workstation's "hosts" file so that your workstation uses the University's Shib QA IdP server rather than the production server. Your SP will direct your browser to shib-idp.umsystem.edu because it is configured in shibboleth2.xml. However, the production Shib servers do not yet know anything about your SP. The change to your workstation hosts file will cause your browser to think it is talking to the production servers when it is, in fact, talking to the QA servers. You do not have to change the hosts file on your SP to test against the QA server!
- 8. Open a command prompt on your workstation using "Run as Administrator".
- 9. Change directories to c:\windows\system32\drivers\etc
- 10. Edit the hosts file and add an entry for the Shib QA server IP address with the production name:

128.206.14.152 shib-idp.umsystem.edu shib-idp-qa.umsystem.edu

The settings above will "trick" your browser into going to the QA server (128.206.14.152) whenever the hostname of shib-idp.umsystem.edu is used. Save the file. When you are finished testing simply comment out the line above in your workstation's hosts file by putting a "#" at the beginning of the line and save the file.

- 11. Close all browser windows. This is necessary because your browser may have cached the hostname/ip address for the production Shibboleth servers. Closing the browser down clears this cache.
- 12. Open your browser and point at your site's "secure" directory. For example:

https://myiissite.umsystem.edu/secure

- 13. You should be immediately directed to the Shibboleth (QA) IdP. It looks identical to production. Check your IIS logs and Shibboleth logs in c:\opt\shibboleth-sp\var\log for errors if you are not directed to the IdP. Correct any errors, restart both IIS and shibd services, close your browser and try again.
- 14. On the QA IdP, sign in using your University ID and password.
- 15. You should be directed back to your SP's "secure" directory (assuming you've set one up).

16. To verify that the IdP is sending the correct user attributes to your site you can check the "session" debug URL on your SP. This URL is automatically generated by the Shib daemon on the SP. It is:

https://[yourservername]/Shibboleth.sso/Session

If you see:

A valid session was not found.

Then you don't have a Shibboleth credential for your site. Check your IIS and Shibboleth logs.

Everything is working if you see something like this:

```
Miscellaneous
Client Address: 128.206.92.200
Identity Provider: https://shib-idp.umsystem.edu/idp/shibboleth
SSO Protocol: urn:oasis:names:tc:SAML:2.0:protocol
Authentication Time: 2012-04-03T15:30:09.252Z
Authentication Context Class:
urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProtectedTransport
Authentication Context Decl: (none)
Session Expiration (barring inactivity): 539 minute(s)
```

```
Attributes
SAM_ACCOUNT_NAME: 1 value(s)
eppn: 1 value(s)
persistent-id: 1 value(s)
```

Note that if you previously set 'showAttributeValues="true"' in your shibboleth2.xml file then you'll see the actual user attribute values, such as your username, passed to your SP rather than a message indicating the values were there.

17. After you finish testing, change your "hosts" file back to normal and let the Shib support team know that everything is working (or not).

2.6.8. Protect Other Resources with Shibboleth

After you have the "secure" location working, you can now expand your IIS/Shibboleth configuration to include other URL "locations" on your web site. To do this you simply add or change the configuration in your shibboleth2.xml file.

For example, if you want to protect your /cgi-bin directory, then add another "Path" tag to your RequestMapper as shown below:

```
<RequestMapper type="Native">
<RequestMap>
<!-- The example requires a session for documents in /secure
on the containing host with http and https on the default
ports. Note that the name and port in the <Host> elements
MUST match Apache's ServerName and Port directives or the
IIS Site name in the <ISAPI> element above. -->
<Host name="myiissite.umsystem.edu">
<Path name="myiissite.umsystem.edu">
<Path name="secure" authType="shibboleth" requireSession="true"/>
<Path name="cgi-bin" authType="shibboleth" requireSession="true"/>
</Host>
</RequestMap>
</RequestMapper>
```

A note about protecting your ENTIRE web site: While it sounds like a good idea to simply protect your entire site, this doesn't often work out as planned. There are certain locations on your site that must be available to users who have not authenticated. One of them is /Shibboleth.sso. This location and everything under it is used by Shibboleth to handle the authentication process. Also, exclude any area of your site that might contain error pages or images you wish served up to users who can't or haven't yet authenticated (don't forget to allow access to your site logo displayed on the IdP). Exclude these locations from your config, or configure them to be explicitly open.

Generally, no changes are required to either your SP metadata or the IdP configuration for you to protect additional URL locations on your web server. Some changes may be required if you add an IP address or a new hostname/URL to your server, for example when you serve up multiple sites from one IIS server.

Appendix A: Setting up a Reverse Proxy to Protect non-IIS Applications

The Shibboleth SP software for windows runs only as an IIS ISAPI filter, therefore you must use IIS to "reverse proxy" any non-IIS web applications you wish to integrate with Shibboleth. This appendix is not a definitive how-to on IIS reverse proxy techniques, but it does document a basic IIS reverse proxy with shibboleth SP setup. The instructions below use a second IIS site (that is not configured with Shibboleth protection) as the "non-IIS" application. This is an example only; there is no need to configure a reverse proxy in IIS to serve up other IIS applications on the same host.

Before you get started it is critical that you understand how your web application gets its user credentials and manages user sessions. Applications that have a hard-coded authentication mechanism probably can't be configured to use Shibboleth. However, many applications have a configurable authentication system that allows the application owner to plug-in his or her own authentication module or code by modifying a particular web page or section of application code.

Typically you would configure your proxy to handle all of your applications' URLs and then set up shibboleth to protect those URLs in the proxy. This is the simplest configuration and the one that will be documented here.

However, it is also possible to have shibboleth protect only a single page or URL location. That page might be a custom login page that your software vendor allows you to modify so that you can provide your own authentication scheme. In that scenario, you would configure the IIS reverse proxy to proxy all pages, but have Shibboleth protect only a single URL by adding a specific file path in the <RequestMap> section of the shibboleth2.xml file. For example:

```
<RequestMap>
<!-- The example requires a session for the /login.jsp page
of my test stie.
-->
<Host name="myiissite.umsystem.edu">
<Path name="/speciallogin.jsp" authType="shibboleth"
requireSession="true"/>
</Host></RequestMap>
```

Regardless of whether you are protecting one page or a whole site, the objective of the reverse proxy configuration is basically the same: to take URLs sent to IIS and transparently forward them on to your application with some indication that the user has authenticated with Shibboleth.

Prerequisites:

- Your non-IIS application must be running on the local machine on a TCP port that is not used for IIS. Typically, application servers like Tomcat or WebSphere are easily configured to run on port 8080. You can run your application on any port you chose, you can also have multiple non-IIS applications running on various different ports and all of them can be proxied simultaneously by IIS.
- Set up IIS to act as a Shibboleth SP as outlined in the document above. Set up a test "secure" directory and build an .asp or .aspx page to display the local environment variables. Test that Shibboleth is working correctly and that you are receiving the attributes you need with IIS and the test "secure" directory. **Do not skip this prerequisite.**
- Verify that you can connect to your non-IIS application. For example, open a browser on your server and navigate to http://127.0.0.1:8080/. You should see your application's default page.
- If possible, create a page in your non-IIS application that does nothing but display environment variables or headers passed to it or find a page that you can use to dump or debug the environment. It would be best to have a page like the one in Appendix C (built for your application environment) that can display the entire server environment.
- Know and understand which of your application's URLs you need protected with Shibboelth. Is it a whole URL path or just one login page?
- Decide what URL your end-users will see in their browser when they use your application, it can be different from the one your application uses. For example, you can map https://myiissite.umsystem.edu/secureBookList/ to your application's /BookList/ URL. Or you can have them match. You can also have your IIS root point at your application root. The "reverse proxy" can handle various combinations of URLs.

Installation and Configuration

For this example, we will be protecting the application page running on http://127.0.0.1:8080/abmsecure/ with a proxy on IIS so that end users see https://myiissite.umsystem.edu/proxytest/.

The /abmsecure/index.asp page will be our non-IIS application and is configured to dump out the application's environment (this is a copy of the sample page in Appendix C).

We'll configure the proxy and shibboleth in two phases. The first phase will be to configure the proxy. The second will be to protect the proxy with Shibboleth.

Phase 1: Configure the proxy.

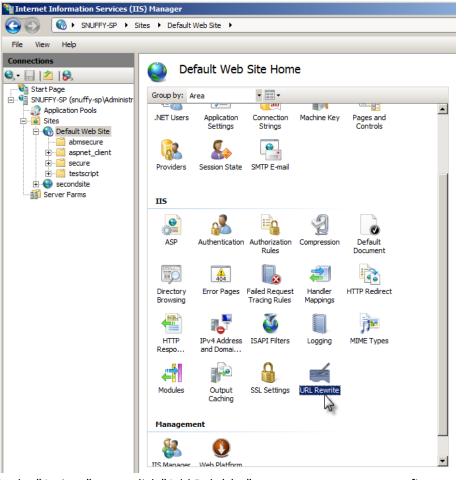
1. Download the Application Request Routing (ARR) Extension for IIS

The Application Request Routing (ARR) extension must be installed using the Microsoft Web Platform Installer. The home page for the utility is http://www.iis.net/download/ApplicationRequestRouting

The ARR extension is required to provide the "reverse proxy" functionality in IIS.

After the ARR extension is installed, restart IIS.

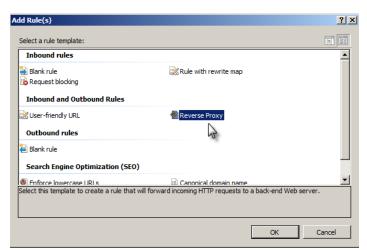
2. Open the IIS Manager and navigate to your default web site (or the site that will be acting as the "reverse proxy"). Double-click the URL Rewrite icon.



3. In the "Actions" pane, click "Add Rule(s)..." to create a new proxy config.

	URL Rewrite	Actions Add Rule(s)
-sp\Administr	Provides rewriting capabilities based on rules for the requested URL address and the content	Revert to Palent
ls	of an HTTP response. Inbound rules that are applied to the requested URL address:	Manage Server Variables View Server Variables

4. In the Add Rule(s) dialog box, click "Reverse Proxy" and then OK.



5. In the "Add Reverse Proxy Rules" dialog box, under the "Inbound Rules" section, enter the server name or IP address of your application. For example if your application is running on port 8080, you would enter 127.0.0.1:8080. This will send requests to your application without them leaving the machine at all. If your application runs without encryption then make sure the "Enable SSL Offloading" checkbox is checked.

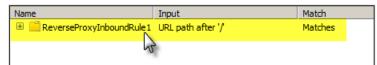
Ad	ld Reverse Proxy Rules	? ×
	Inbound Rules Enter the server name or the IP address where HTTP requests will be forwarded: 127.0.0.1:8080	1
	Example: contentserver1	
	Enable SSL Offloading Selecting this option will forward all HTTPS requests over HTTP.	

 In the "Outbound Rules" section the IP address:port (or machine name) will be repeated in the "From:" line. Change the "To:" line to the URL your users will see. For example: myiissite.umsystem.edu. Click OK.

Responses tha	is that use internal dom	is in HTTP responses plications that are behind a ain names. These links mus	
From:			
127.0.0.1:808	30		
Example: cont	entserver1		
To:			
myiissite.umsy	/stem.edu		-
Example: www	v.contoso.com		
			OK Cancel

You should now see the main URL Rewrite pane with one inbound and one outbound rule.

7. Double-click the inbound rule to edit it.



By default the add rules wizard will only set up the rule to proxy an entire site so that anything sent to https://myiissite.umsystem.edu/ will be forwarded to http://127.0.0.1:8080/. If you have other applications or URLs that do not need to be forwarded, then you must modify this very generic rule to protect specific directories.

8. In the Match URL section make sure that the Requested URL is set to "Matches the Pattern" and the Using dropdown is set to "Regular Expressions". For the pattern, we will be changing https://myiissite.umsystem.edu/proxytest to 127.0.0.1:8080/abmsecure, to "^proxytest/(.*)" [don't include the double-quotes]. This regular expression says match a URL starting with proxytest followed by slash and anything else.

Match URL	۲	
Requested URL:	Using:	
Matches the Pattern	Regular Expressions	
Pattern:		
^proxytest/(.*)	Test pattern	
☑ Ignore case		
eave the Conditions and Ser	ver Variables sections empty.	
Conditions		
Server Variables		V

In the Action section make sure the Action Type is "Rewrite" and change the Action Properties URL to http://127.0.0.1:8080/abmsecure/{R:1}. Make sure the "Append query string" checkbox is checked, "Log rewritten URL" is not checked, and "Stop processing of subsequent rules" is checked. Use the Test pattern button to test other regular expressions as needed.

Action	٢
Action type:	
Rewrite	
Action Properties	
Rewrite URL:	
http://127.0.0.1:8080/abmsecure/{R:1}	
Append query string	
Log rewritten URL	

Click "Apply" in the Actions section to save the rule. Then click "Back to rules" to return to the rules list.

Actions		
🔶 Back to Rules		

9. In the outbound rules section, we need two rules. One to handle complete URLs sent back from your application and one to handle relative path URLs. Double click the "ReverseProxyOutboundRule1" to edit it. This is created by default to handle full URL

tra	nslation.						
Out	bound rules that are applied to t	he headers or the o	ontent of an HTTP res	ponse:			
Na	me	Input	Match	Pattern	Action Type	Action Value	Stop
÷	ReverseProxyOutboundR	A, Form, Img	Matches	^http(s)?://127	Rewrite	http{R:1}://myi	False
	ange the Pattern: sponse	to ^http(s))://127.0.0.1	L:8080/abm	secure/(.*)		
	tch the content within: Form, Img	¥	Custom tags:				
_	ntent: atches the Pattern	¥	Using: Regular Expres	sions	T		
_	ttern: http(s)?://127.0.0.1:8080/abmsecur	e/(.*)		Test patte	m		
	Ignore case					_	
Ch	ange the Action P	roperties v	alue to: http	o{R:1}://myi	issite.umsy	stem.edu/p	proxytest/{R:2}
Act	ion						
Ac	tion type:						
R	ewrite						
	ction Properties						
V	alue:						

□ Stop processing of subsequent rules

This will change any URL embedded in your application's HTML so that it is translated from 127.0.0.1/abmsecure/ to myiissite.umsystem.edu/proxytest/.

Click "Apply" and then "Back to Rules" to return to the rules list.

http{R:1}://myiissite.umsystem.edu/proxytest/{R:2}

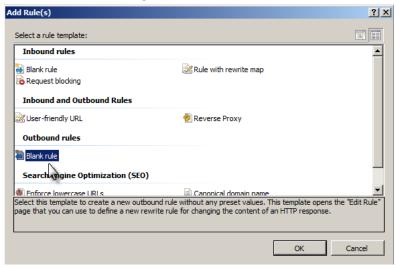


10. Now we must add a second outbound rule to handle relative paths generated by the application. Click "Add Rule(s)..."





11. In the Add Rule(s) dialog, scroll down to the "Outbound rules" section and click "Blank rule".



Click OK.

12. Name the rule "RewriteRelativePaths". Set the Precondition to "ResponseIsHtml1.

Edit Outbound Rule		
Name:		
RewriteRelativePaths		
Precondition:		
ResponseIsHtml1	-	Edit

In the Match section make sure the matching scope is "Response", make sure the "Match the content within:" dropdown box has all the boxes checked except "Use Custom Tags". Under Content: choose "Matches the Pattern". Under Using: choose "Regular Expressions". For the pattern, enter (.*)/(.*) and make sure the "Ignore case" checkbox is checked.

Match	e
Matching scope:	
Match the content within:	Custom tags:
A, Area, Base, Form, Frame, Head, IFrame	
Content:	Using:
Matches the Pattern	Regular Expressions
Pattern:	
^(.*)/(.*)	Test pattern
☑ Ignore case	

Scroll down to the Action section and make sure the Action Type is "Rewrite". Under Action Properties set the Value to "/proxytest/{R:2}".

Action	
Action type: Rewrite	
Action Properties	
Value:	
<pre>/proxytest/{R:2}</pre>	

Click "Apply" and "Back to Rules".



After you've saved your rules, you can open them with a text editor. They are stored in your site's root directory in the web.config file. You may edit the web.config file with a text editor if you don't like the GUI. The resulting XML should look something like this:

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
 <system.webServer>
  <rewrite>
   <rules>
    <rule name="ReverseProxyInboundRule1" stopProcessing="true">
     <match url="^proxytest/(.*)" />
     <action type="Rewrite"
         url="http://127.0.0.1:8080/abmsecure/{R:1}" />
    </rule>
   </rules>
   <outboundRules>
    <rule name="ReverseProxyOutboundRule1"
       preCondition="ResponseIsHtml1">
     <match filterByTags="A, Area, Base, Form, Frame, Head, IFrame,
Img, Input, Link, Script"
      pattern="^http(s)?://127.0.0.1:8080/abmsecure/(.*)" />
      <action type="Rewrite" value="/proxytest/{R:2}" />
     </rule>
     <rule name="RewriteRelativePaths" preCondition="ResponseIsHtml1">
      <match filterByTags="A, Area, Base, Form, Frame, Head, IFrame,
Img, Input, Link, Script" pattern="^(.*)/(.*)" negate="false" />
       <action type="Rewrite" value="/proxytest/{R:2}" />
    </rule>
    <preConditions>
     <preCondition name="ResponseIsHtml1">
```

13. Open a browser on your sp and goto the page using your application proxy URL which will dump out your environment variables. In this example the URL is

https://myiissite.umsystem.edu/proxytest/index.asp

erve 6	r Variables - Windows In	ternet Explorer
GO - (🕘 https://s	HTTP Server Variables
HTTPS_SE	ERVER_SUBJECT	
INSTANC	E_ID	2
INSTANC	E_META_PATH	/LM/W3SVC/2
LOCAL_A	DDR	127.0.0.1
LOGON_U	JSER	
PATH_INF	FO	/abmsecure/index.asp
PATH_TR	ANSLATED	C:\inetpub\secondsite\abmsecure\index.asp
QUERY_S	TRING	
REMOTE_	ADDR	127.0.0.1
REMOTE_	HOST	127.0.0.1
REMOTE_	USER	
REQUEST	_METHOD	GET
SCRIPT_N	IAME	/abmsecure/index.asp
SERVER_1	NAME	127.0.0.1
SERVER_I	PORT	8080
SERVER_I	PORT_SECURE	0
SERVER_I	PROTOCOL	HTTP/1.1
SERVER_S	SOFTWARE	Microsoft-IIS/7.0
URL		/abmsecure/index.asp
HTTP_CO	NNECTION	Keep-Alive
HTTP_AC	CEPT	text/html, application/xhtml+xml, */*
HTTP_AC	CEPT_ENCODING	gzip, deflate
HTTP_AC	CEPT_LANGUAGE	en-us
HTTP_HO	ST	127.0.0.1:8080
HTTP_MA	X_FORWARDS	10
HTTP_USE	ER_AGENT	Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.0; Trident/5.0)
	ORIGINAL_URL	/proxytest/index.asp
HTTP_X_F	FORWARDED_FOR	192.168.120.129:50271
HTTP_X_A	ARR_SSL	[100] S. M. M. S. Marson, J. S. Manna, N. Sanara, N. Sanara, S. S. Manna, S. S. Manna, S. Sanara, S. S. Manna, S. Sanara, and S. Sanara, S. S. Manna, S. S. Sanara, S. Sanara, S. Sanara, S. Sanara, S. Sanara, S. S. Sanara, Sanara, S. Sanara, S. Sanara, S. Sanara, Sana Sanara, Sanara, San
HTTP_X_A	ARR_LOG_ID	b88df933-2529-44ff-966c-a675332edbcf

If everything is configured correctly, you should see your page dump out its environment variables. Notice the HTTP_X_... variables. Those indicate that the IIS proxy has forwarded the request to your application.

Because we haven't configured the Shibboleth ISAPI module to protect the URL you should see standard variables. Obviously, if you are protecting an application page that expects an authenticated session, you should see an authentication error.

Phase 2: Protect the URL with Shibboleth.

In this phase, we'll simply configure the Shibboleth SP software to protect the /proxytest URL under IIS.

1. Open your shibboleth2.xml file and add the following proxytest path line to your <Host> config in the <RequestMap> section.

```
<Host name="myiissite.umsystem.edu">
<Path name="secure" authType="shibboleth" requireSession="true"/>
<Path name="proxytest" authType="shibboleth" requireSession="true"/>
</Host>
```

Save the shibboleth2.xml file and restart the shibboleth service and then restart IIS.

2. Close your browser, re-open it and navigate to your protected URL (should be the same as above) which is now protected with shibboleth.

It should send you to the IdP for authentication and when you return your page should display a new set of environment variables that include the attributes sent from the IdP.

HTTP Server Variables - Windows Internet Explore	ETTP Server Variables - Windows Internet Explorer		
🕒 🕤 🗢 🧭 https://s 👫 a anning ann ann /pro	xytest/i 🍳 🗹 Certific 🗟 😏 🗙 🔏 HTTP Server Variables 🗙		
SERVER_PROTOCOL	HTTP/1.1		
SERVER_SOFTWARE	Microsoft-IIS/7.0		
URL	/abmsecure/index.asp		
HTTP_CACHE_CONTROL	no-cache		
HTTP_CONNECTION	Keep-Alive		
HTTP_ACCEPT	text/html, application/xhtml+xml, */*		
HTTP_ACCEPT_ENCODING	gzip, deflate		
HTTP_ACCEPT_LANGUAGE	en-us		
HTTP_COOKIE	_shibstate_29002348=https%3A%2F%2F %2F _shibsession_64656661756c7468747470733a2f2f736e7566667973702e61		
HTTP_HOST	127.0.0.1:8080		
HTTP_MAX_FORWARDS	10		
HTTP_REFERER	https://://idp/profile/SAML2/Redirect/SSO		
HTTP_USER_AGENT	Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.0; Trident/5.0)		
HTTP_SHIBAPPLICATIONID	default		
HTTP_SHIBSESSIONID	_ccd50310caf17a347600bd2ca8845c07		
HTTP_SHIBIDENTITYPROVIDER	https://www.https/https//www.https://www.https://www.https://www.https://www.https://www.https://www.https://www.https://www.https://www.https/https//www.https/https//www.https//www.https/https//www.https//www.https//www		
HTTP_SHIBAUTHENTICATIONINSTANT	2012-05-03T20:19:02.197Z		
HTTP_SHIBAUTHENTICATIONMETHOD	urn:oasis:names.tc:SAML:2.0:ac:classes:PasswordProtectedTransport		
HTTP_SHIBAUTHNCONTEXTCLASS	urn:oasis:names.tc:SAML:2.0:ac:classes:PasswordProtectedTransport		
HTTP_SHIBSESSIONINDEX	6e282285f091506a561ba04a94efbe0d5fd96ae46cdbcffdf9b9aa79d13198e2		
HTTP_DISPLAYNAME	and the second		
HTTP_MEMBEROF	CN=MU DoIT TigerNet1X Wireless Users, OU=Distribution Groups, OU=IS.		
HTTP_SAMACCOUNTNAME	install http://www.initia		
HTTP_REMOTEUSER	and the state of the second		
HTTP_X_ORIGINAL_URL	/proxytest/index.asp		
HTTP_X_FORWARDED_FOR	192.168.120.129:50293		
HTTP_X_ARR_SSL	Traffic BET-1775 B-Millionali (1-17-ababila). Trafficial Mathematical (1775 Productor Mathematics and 1797 marking and transmission and Productor (1		
HTTP_X_ARR_LOG_ID	01c4a20a-5f20-48db-bc44-7924dcce908e		

3. Modify your configuration to protect your entire site, or just a "login" page that sets up your application's session.

Appendix B: Setting up Metadata to Handle Multiple Hostnames

It is often necessary for a web server to serve up the same applications on different host names. What the application does and what attributes it receives in this scenario are identical, just the hostname changes.

For example, if you have a load balancer in front of your web server, the load balancer may have one name (myiissite.umsystem.edu), but your individual server will have another (myiissite-01.umsystem.edu). To test a server before adding it to the load balancer, you may want to point your browser directly at your server instead of through the loadbalancer.

Shibboleth can handle this, but you must **manually** edit your metadata file to allow for the different hostnames and each hostname must be in the metadata file. There is currently no way to specify multiple hostnames in the shibboleth2.xml file.

In your SP's Metadata file near the bottom, you will see multiple "<md:AssertionConsumerService" tags. Each of those tags must be repeated for all the different hostnames your server will respond to. The IdP uses these tags to verify where to send the attributes for your SP's entityID.

To add multiple host names simply copy-and-paste the existing block of "<md:AssertionConsumerService" tags. You must then do TWO things on each copied line: 1) change the "Location=" hostnames to your desired hostname and 2) change the "index=" values so that there are no duplicates. **Double check your index values!**

An example is shown below:

```
index="3"/>
    <md:AssertionConsumerService
Binding="urn:oasis:names:tc:SAML:1.0:profiles:browser-post"
Location="https://myiissite.umsystem.edu/Shibboleth.sso/SAML/POST"
index="4"/>
    <md:AssertionConsumerService
Binding="urn:oasis:names:tc:SAML:1.0:profiles:artifact-01"
Location="https://myiissite.umsystem.edu/Shibboleth.sso/SAML/Artifact"
index="5"/>
<!-- New host names for this server -->
<!-- Changed hostname from myiissite to myiissite-01 and changed indexes! -->
<md:AssertionConsumerService
Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST"
Location="https://myiissite-01.umsystem.edu/Shibboleth.sso/SAML2/POST"
index="6"/>
    <md:AssertionConsumerService
Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST-SimpleSign"
Location="https://myiissite-01.umsystem.edu/Shibboleth.sso/SAML2/POST-
SimpleSign" index="7"/>
    <md:AssertionConsumerService
Binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Artifact"
Location="https://myiissite-01.umsystem.edu/Shibboleth.sso/SAML2/Artifact"
index="8"/>
    <md:AssertionConsumerService
Binding="urn:oasis:names:tc:SAML:2.0:bindings:PAOS"
Location="https://myiissite-01.umsystem.edu/Shibboleth.sso/SAML2/ECP"
index="9"/>
    <md:AssertionConsumerService
Binding="urn:oasis:names:tc:SAML:1.0:profiles:browser-post"
Location="https://myiissite-01.umsystem.edu/Shibboleth.sso/SAML/POST"
index="10"/>
    <md:AssertionConsumerService
Binding="urn:oasis:names:tc:SAML:1.0:profiles:artifact-01"
Location="https://myiissite-01.umsystem.edu/Shibboleth.sso/SAML/Artifact"
index="11"/>
```

Appendix C: Sample IIS Page to Display Shibboleth Attributes

The code below can be used to help debug a Shibboleth SP protected directory or URL. Place it in a file called index.asp at the root of your protected directory, e.g. c:\inetpub\wwwroot\secure\index.asp. Then call the page via your site, e.g. https://myiissite.umsystem.edu/secure/index.asp. You will no doubt have to change the Request.ServerVariables section to match the variables you've configured in your attribute-map.xml. This code comes from the Internet2 Shibboleth InstallFest virtual machine.

```
<%@ EnableSessionState=False%>
<%
' index.asp
' dumps http server variables
응>
<HTML>
<HEAD><TITLE>HTTP Server Variables</TITLE></HEAD>
<BODY BGCOLOR=#FFFFFF>
<BR>
<H2 align="center">Shibboleth Test Page</H2>
<BR><BR>
<I>Authenticated User Information </I>(information received from your IdP, if
any, will be populated below):<BR><BR>
<TABLE>
<TR><TD><B>First name:</B></TD><TD><INPUT type="text"
value="<%=Request.ServerVariables("HTTP_GIVENNAME")%>"></TD></TR>
<TR><TD><B>Last name:</B></TD><TD><INPUT type="text"
value="<%=Request.ServerVariables("HTTP_SN")%>"></TD></TR>
<TR><TD><B>Email address:</B></TD><TD><INPUT type="text"
value="<%=Request.ServerVariables("HTTP_MAIL")%>"></TD></TR>
<TR><TD><B>eduPersonPrincipalName:</B></TD><TD><INPUT type="text"
value="<%=Request.ServerVariables("HTTP_EPPN")%>"></TD></TR>
<TR><TD><B>REMOTE_USER:</B></TD><TD><INPUT type="text"
value="<%=Request.ServerVariables("HTTP_REMOTEUSER")%>"></TD></TR>
</TABLE>
<BR><BR><BR>
<HR>
<H2>Raw HTTP Server Variables</H2>
<TABLE BORDER=1>
<TR><TD VALIGN=TOP><B>Variable</B></TD><TD VALIGN=TOP><B>Value</B></TD></TR>
<% For Each key in Request.ServerVariables %>
   <%If Trim(key) <> "HTTP_SHIBSPOOFCHECK" Then%>
 <TR>
 <TD><% = key %></TD>
```

```
<TD>
 <%
 if Request.ServerVariables(key) = "" Then
 Response.Write "&nbsp" ' To force border around table cell
 else
 Response.Write Request.ServerVariables(key)
 end if
Response.Write "</TD>"
 응>
 </TR>
    <%End IF
Next
응>
</TABLE>
<BR>
<BR>
</BODY>
</HTML>
```