

SECURING GROUPWISE 6.5 END TO END WITH SSL

BY FRANK GIOFFRE AND TAY KRATZER



There are many reasons for enhancing your security posture, such as security compliance, sensitivity of the data, and the use of public networks.

A majority of today's correspondence is electronic in nature with e-mail ranking first and faxes ranking second. Real "pen and paper" correspondence is becoming extinct. It is extremely unlikely that any corporation can avoid sending or accessing sensitive or confidential information over some sort of electronic medium. This raises security concerns about the safety of your data. In this article we explain the need for security as well as the means by which to implement a high-security solution.

THE NEED FOR ENHANCED SECURITY

There are many reasons for enhancing your security posture, such as security compliance, sensitivity of the data, and the use of public networks. No matter what the reason, there is a general consensus that our systems all need to be more secure than they are today.

Regulations are one of the driving forces behind the latest push for increased security. If you work in the health care industry you know what kind of stringent requirements HIPAA has placed on all of your systems. Other industries are also under extreme pressure to secure all data systems. You need to be able to implement, document and prove the integrity of your systems.

Every day we all see stories about corporate espionage, hacking or destruction of corporate data. You may be so paranoid by now that you want to do everything necessary to make your e-mail system as secure as humanly possible. You have already taken one step in the right direction by using GroupWise 6.5 as it is the most secure messaging system available.

CERTIFICATE AUTHORITIES AND PKI

In order for a server-based public key (See *The SSL Architecture* on page 54.) to be considered valid it must be signed by a "certificate authority." A certificate authority is an authoritative

entity that has ratified a particular host's public key or identity. A certificate authority can be likened to a government agency which issues a document that verifies the identity of a person, e.g., a driver's license or a passport. Those documents can be used by a Notary Public to verify the identity of the person whose signature they're witnessing.

The certificate authority adds its own signature to a host's public key which ratifies the public key. A server-based public key cannot be used until it has been signed by a certificate authority. When the public key has been signed by a certificate authority it is commonly called a certificate. The private key is then called the key. The private key may be secured via a password embedded within the private key. The public key does not need to be secured because it is intended for public distribution. (To create a server-based public and private key, see *Get a Certificate* on page 46.)

FOLLOWING IS HOW SECURE COMMUNICATION PROCEEDS WHEN THE PKI COMPONENTS ARE IN PLACE ON A SERVER:

- The client host requests a session with the server host.
- The server host requests that all data being sent to the server be sent using the Public Key.
- The server host acquires access to its Private Key, which may have been stored in a password-protected file.
- The server host receives the data which has been encrypted with its Public Key.
- The server host decrypts the encrypted data with its Private Key.

CERTIFICATE AUTHORITIES AND PKI IN A NOVELL ENVIRONMENT

Most customers have the trappings of both a Certificate Authority

For more information on Certificate Authorities and PKI in a Novell environment, visit <http://developer.novell.com/research/appnotes/2002/may/01/a020501.htm>.

and the ability to generate public and private keys within their environment. With eDirectory 8.6 or better and GroupWise 6.5 you have all the software to:

- Create a Public/Private Key pair.
- Create a Certificate Signing Request.
- Create a Certificate.

The Novell International Cryptographic Infrastructure (NICI) API allows software to retrieve the PKI key material (private keys), passwords for those private keys and certificates from eDirectory. The key material is stored as an object called a Key Material Object (KMO). Every NetWare 5.1 or higher server automatically has two Key Material Objects created specific to that server. They are usually located in the same eDirectory OU (or context) as the server and they are named: "SSL CertificateDNS - <server name>" and "SSL CertificateIP - <server name>".

Software such as NetWare Remote Manager, iMonitor, and Apache Web server are designed to use NICI to read Key Material Objects out of eDirectory. However, GroupWise agents are not designed to retrieve their keys and certificates from NICI and eDirectory. So although eDirectory, NICI and the Novell PKI infrastructure are used in the process of creating certificates, the certificates are not retrieved from eDirectory and must be exported to a file-based format. The advantage to this architecture is that GroupWise agents can still use their keys and certificates even if eDirectory is unavailable for some reason (for example, gateway servers, like GWIA in particular, need to be available as much of the time as possible).

SECURING GROUPWISE END TO END

How secure is GroupWise and do you need even more security? The answer depends on how you are using GroupWise. In GroupWise 6.5, client to POA connections may be protected either by the traditional proprietary encryption method or by SSL, which is explained later. The connection between the GroupWise client and the POA (Post Office Agent) is always encrypted using a proprietary encryption method. Other connections like the one between your Web browser and WebAccess may or may not be encrypted depending on your settings. We will examine every connection and show you how to encrypt all of them using SSL. Please note that these SSL-enabling features are all available in GroupWise 6.5. Older versions of GroupWise will have some, but not all of these encryption capabilities.

Some GroupWise systems today don't take advantage of these features because either the GroupWise administrator isn't aware that the features exist or they think they are difficult to configure. Not so! You'll be able to take these simple steps and quickly have a more secure GroupWise system in less than one hour!

PREREQUISITES

Before beginning, be sure to fulfill the prerequisites below:

- System
- GroupWise 6.5
- eDirectory 8.6
- Administrator's Workstation

If you haven't already done so, install the Novell International Cryptographic Infrastructure for Windows on the ConsoleOne machine. This software helps the Novell Certificate Server Snap-in (installed later) to function properly.

Some GroupWise systems today don't take advantage of these features because either the GroupWise administrator isn't aware that the features exist or they think they are difficult to configure. Not so! You'll be able to take these simple steps and quickly have a more secure GroupWise system in less than one hour!

- Go to <http://download.novell.com>.
- Select [Novell International Cryptographic Infrastructure] and the [Windows 2000] platform. Then select [<All Dates>] and the [English] language and then select the "Submit Search" icon.
- Save the resulting .exe file to your hard drive and execute it and follow the Installation Wizard instructions.

NOVELL CERTIFICATE CONSOLE (PKI)

SNAP-INS FOR CONSOLEONE

Although you have a CA and a PKI infrastructure already in place, you might not have the Novell Certificate Console (PKI) Snap-ins for ConsoleOne installed. Determine this by loading ConsoleOne and selecting Help | About Snap-ins. Make sure you have the Novell Certificate Server Snap-in version 2.21 or better. If you don't have them, download them from the Novell Web site and install them now using the following steps:

- Visit <http://download.novell.com>.
- Download the latest PKI Snap-ins for ConsoleOne by selecting [Certificate Server], platform [Windows 2000], choosing [<All Dates>] and the [English] language. Then select the Submit Search icon.
- Save the resulting .zip file to your hard drive.
- Unzip that file to the ConsoleOne folder, such as C:\NOVELL\CONSOLEONE\1.2. Be sure to indicate the full path to the 1.2 directory.

GET A CERTIFICATE

One common item that you will need for all agents running on the same server is a security certificate (commonly referred to as a

cert). The certificate is issued by a CA (Certificate Authority) and is based on information that you provide to the CA. To receive a certificate you must first provide a CSR (Certificate Signing Request). All SSL-capable systems have a procedure that allows you to generate a CSR. GroupWise has the GroupWise CSR Generation (GWCSRGEN) Utility.

GWCSRGEN can generate a CSR and private key file. You then send the CSR to a third-party CA and wait for your certificate. When there is no interaction with external GroupWise domains (such as POA to POA communications), save some money by minting your own certificates. We'll follow that procedure here.

STEP 1 CREATE THE CSR

Run GWCSRGEN.EXE which is found in the Software Distribution Directory (SDD) in the directory ADMIN\UTILITY\GWCSRGEN. Complete the information on the dialog. (See Figure 1.) Some things to remember: Choose a Key filename that makes sense to you and use a .KEY extension for the Key filename and a .CSR extension for the CSR filename. Your password for the Key is case-sensitive. Use the official 2-letter abbreviation for your country. Enter your organization's full name, the division of your organization that is requesting this certificate and the DNS name of the host server that will use this certificate.

STEP 2 CREATE THE CERTIFICATE USING THE EDIRECTORY CA SERVER

The certificate is created or minted by the CA using ConsoleOne. We'll use the CSR that we generated earlier.

- In ConsoleOne, highlight a container in the tree. It doesn't matter which container you choose, but it ought to be a

FIGURE 1

Running the GroupWise CSR Generation utility (GWCSRGEN.EXE) allows you to generate a Certificate Signing Request (CSR).



FIGURE 2

The Issue Certificate dialog allows you to choose various options for your certificate. GroupWise agents must have both Data Encipherment and Key Encipherment to properly load the certificates.



- container in your organization, and probably should be the container where your server is located in the tree.
- From the menu, select Tools and then select Issue Certificate.
- In the Filename box, enter the name of the CSR you just created and select Next.
- On the next screen specify Organizational Certificate Authority and select Next.
- On the Key Type screen specify Custom for the type. Under Key Usage, select all three options.

This is very important as GroupWise agents must have both Data Encipherment and Key Encipherment. If you don't pick the proper options, the agents will not properly load the certificates (Error 8209). (See Figure 2.)

- Click Next to continue.
- Specify your certificate validity period. For GroupWise agents, you may want to make this longer than 1 year.
- On the Summary screen, check your information and select Finish.
- On the Save Certificate screen, select File in Base64 format and specify a certificate filename. Use the default extension of .b64. Don't use filenames longer than 8 characters for any certificates or key names.

STEP 3 USE THE CERTIFICATE

Save the certificate and the key file on the file server because the GroupWise agents must be able to access the files when they start up. Be sure that there are no directory names longer than 8 characters anywhere in the path. Any GroupWise agent running on the specified server can now use the certificate.

SECURING COMMUNICATIONS BETWEEN THE GROUPWISE CLIENT AND THE POA

The GroupWise 6.5 client and the GroupWise 6.5 POA can communicate securely via SSL. Older GroupWise clients cannot communicate with the POA via SSL. Setting up this kind of communication requires no action on the GroupWise client. The only steps you must take are with the GroupWise POA:

STEP 1 ADD THE CERTIFICATE TO THE POA

- From ConsoleOne, bring up the properties of the POA.
- From the GroupWise tab select SSL Settings.
- Fill in the fields, specifying the certificate file and key file.
- Click on Set Password and enter the case-sensitive password you chose when you created the key file.
- Select Apply to save the POA properties.

STEP 2 ENABLE SSL COMMUNICATIONS ON THE POA

- From the GroupWise tab of the POA, select Network Address.
- Select SSL Enabled for Local Intranet Client/Server.
- Select SSL Enabled for Internet Proxy Client/Server.
- Click OK to save changes and exit the POA properties screen.

NOTE: Both of the fields "Local Intranet Client/Server" and "Internet Proxy Client/Server" have an "SSL Required" choice. If you choose "SSL Required," then you effectively disable all access from all non-GroupWise 6.5 clients and any GroupWise WebAccess Agents that don't have SSL enabled.

STEP 3 TEST FOR SSL SUPPORT

- Log in to the GroupWise 6.5 client.

FIGURE 3

When SSL is enabled, the GroupWise 6.5 client shows that SSL is enabled with a padlock icon in the lower right-hand corner of the interface.



- Confirm that you see a padlock icon in the lower right-hand corner of the GroupWise 6.5 client. (See Figure 3.)

SECURING COMMUNICATIONS BETWEEN THE GROUPWISE WEBACCESS AGENT AND THE POA

To secure communications between the GroupWise WebAccess Agent and the POA, take each step in the previous section called *Securing Communications Between The GroupWise Client and The POA* and then take the following additional steps.

STEP 1 ADD THE CERTIFICATE TO THE WEBACCESS AGENT

- From ConsoleOne, bring up the properties of the WebAccess Agent.
- From the GroupWise tab select SSL Settings.
- Fill in the fields, specifying the certificate file and key file.
- Click on Set Password and enter the case-sensitive password.
- Click Apply to save the WebAccess Agent properties.

STEP 2 ENABLE SSL COMMUNICATIONS ON THE WEBACCESS AGENT

- From the GroupWise tab of the WebAccess Agent, select Network Address.
- Select SSL Enabled for Client/Server.
- Click OK to save changes and exit the WebAccess Agent properties screen.
- Bring the WebAccess Agent down and back up again so that it loads the SSL configuration settings.

STEP 3 TEST FOR SSL SUPPORT

- Configure your WebAccess Agent so that the logging level is set to verbose. Do this from the WebAccess Agent console screen if needed by selecting F10 | Logging Options.

FIGURE 4

The WebAccess Agent log file should be similar to this log file.



- Log in to GroupWise WebAccess. Be sure to log in as a user that is accessing the WebAccess Agent and the POA that have SSL enabled.
- The log file of the WebAccess Agent should show something like "DOMAIN.POSTOFFICE.USERID User connected via SSL." (See Figure 4.)

SECURING COMMUNICATIONS BETWEEN POA AND MTA

For communications between the POA and MTA (Message Transfer Agent) to be secured, set up SSL for both the POA and the MTA.

STEP 1 ADD THE CERTIFICATE TO THE POA

- From ConsoleOne, bring up the properties of the POA.
- From the GroupWise tab select SSL Settings.
- Fill in the fields, specifying the certificate file and key file. (See Figure 5.)
- Click on Set Password and enter the case-sensitive password.
- Click Apply to save the POA properties.

STEP 2 ENABLE SSL COMMUNICATIONS ON THE POA

- From the GroupWise tab of the POA, select Network Address.
- Select Enabled under SSL for Message Transfer. (See Figure 6.)
- Click OK to save changes and exit the POA properties screen.

STEP 3 ADD THE CERTIFICATE TO THE MTA

- From ConsoleOne, bring up the properties of the MTA.
- From the GroupWise tab select SSL Settings.
- Fill in the fields, specifying the certificate file and key file.
- Click on Set Password and enter the case-sensitive password.
- Click Apply to save the MTA properties.

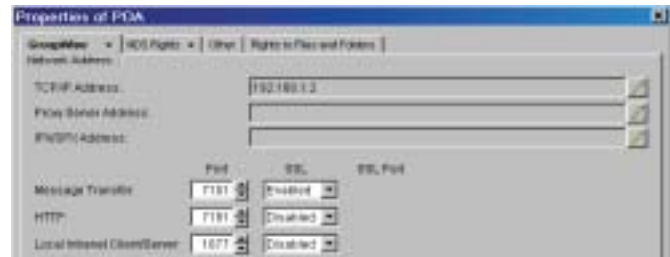
FIGURE 5

On the SSL Settings property page of the MTA, indicate the path to the Certificate and the Key. These files are used to encrypt MTA to MTA and MTA to POA communication.



FIGURE 6

On this POA Properties dialog, the POA has been configured to communicate via SSL to the MTA.



STEP 4 ENABLE SSL COMMUNICATIONS ON THE MTA

- From the GroupWise tab of the MTA, select Network Address.
- Select Enabled under SSL for Message Transfer and click OK. (See Figure 7.)
- Restart both the MTA and the POA

Check the settings by looking at the startup section of the POA log file. You should see an entry that reads as Message Transfer over SSL: Enabled. (See Figure 8.)

SECURING COMMUNICATIONS MTA TO MTA

For communications between MTAs (Message Transfer Agents) to be secured, set up SSL for all subject MTAs. The good news is that if you already followed the steps given above on all your MTAs then you are already done. If you have other MTAs that need to be secured, simply follow these steps for each additional MTA:

- Copy your certificate file and key file to the server running the MTA. Use a secure transfer protocol, like https or secure ftp to copy key files between machines. Always ensure that the key file is protected from being read from the place where it's stored on the file system by anyone other than the agents configured to use it.
- Add the certificate to the MTA as presented in Step 3 of Securing Communications Between POA and MTA.
- Enable SSL communications on the MTA as presented in Step 4 of Securing Communications Between POA and MTA.
- Restart the MTA.

SECURING GWIA (GROUPWISE INTERNET AGENT)

SESSIONS WITH STARTTLS

With the advent of Extended SMTP (eSMTP), we now have the capability of securing Internet mail connections by using SSL or Transport Layer Security (TLS). GWIA 6.5 supports SSL and TLS, and the setup is just as simple as it is for the other agents. The only issue here is that you only have control over your end of the SMTP transaction. Both SMTP servers (yours and the other end) must support TLS for messages to get encrypted. Once set up, your GWIA will send all messages using TLS if the other host supports the protocol; otherwise, the message will be sent using plain SMTP.

STEP 1 ADD THE CERTIFICATE TO GWIA

- You already have minted a certificate and it is running on the GWIA server.
- From ConsoleOne, bring up the properties of GWIA.
- From the GroupWise tab select SSL Settings.
- Complete the fields, specifying the certificate file and key file. (See Figure 5.)
- Click on Set Password and enter the case-sensitive password.
- Click Apply to save the GWIA properties.

STEP 2 ENABLE SSL COMMUNICATIONS FOR GWIA

- From the GroupWise tab for GWIA, select Network Address.
- Select Enabled under SSL for SMTP. (See Figure 9.)
- Click OK to save the changes and exit the GWIA properties screen.
- Restart GWIA.

STEP 3 TEST FOR TLS SUPPORT

Use any TELNET client you have to verify that your GWIA or

FIGURE 7

On this MTA Properties dialog, communication between MTAs is configured as SSL over port 7100.

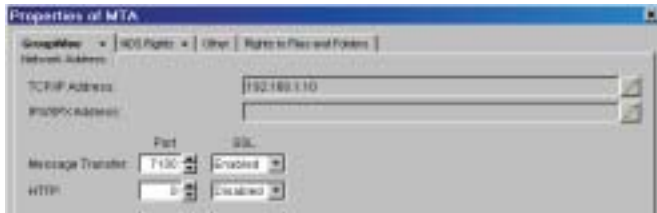
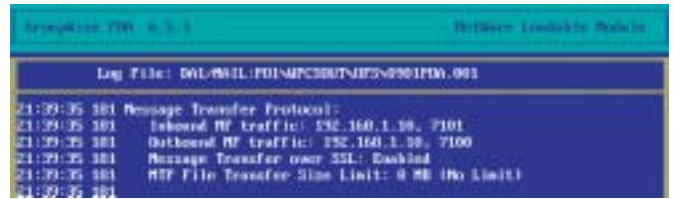


FIGURE 8

When SSL is enabled and functioning, you can confirm from the POA Log Screen that SSL communication is enabled on the line reading "Message Transfer over SSL: Enabled."



any other SMTP host supports TLS. These steps use the Windows 2000 standard TELNET client:

- Start the TELNET client from a DOS Command Prompt window and issue the command TELNET.
- From the Microsoft Telnet> prompt, type OPEN DA1.digitalairlines.com 25 (replace DA1.digitalairlines.com with the registered host name of the SMTP server) and press Enter.
- After a few seconds you should get a response with the identity of the host. Then issue the command EHLO.

Depending on your TELNET settings, the command you type may not be visible on the screen.

- If you get the response message "250-STARTTLS," then the host supports TLS. (See Figure 10.)

SECURING HTTP MONITORING SESSIONS

All of the GroupWise 6.5 agents and gateways are instrumented so that you can access statistics and configuration information with just a Web browser. In the default configuration these sessions use standard HTML over port 80, which means that the data crosses the wire completely unencrypted. You can secure these sessions by turning on SSL support. All of the agents are configured in the same way so we'll describe the process only once.

STEP 1 VERIFY HTTP FUNCTIONALITY

Before setting up SSL, verify the proper operation of the HTTP monitoring by pointing a Web browser at the proper IP address and port of the agent. This **MUST WORK** before attempting to secure the connection with SSL. If it doesn't, fix it before proceeding.

STEP 2 ADD THE CERTIFICATE TO THE AGENT

Perform these steps only if this agent has not yet had a certificate associated with it.

- You should have already minted a certificate and placed it on the server where the agent is running.
- From ConsoleOne, bring up the properties of the agent or gateway.
- From the GroupWise tab select SSL Settings.
- Fill in the fields, specifying the certificate file and key file. (See Figure 5.)
- Click on Set Password and enter the case-sensitive password.
- Click Apply to save the agent properties.

STEP 3 ENABLE SSL COMMUNICATIONS FOR THE AGENT

- From the GroupWise tab for the agent, select Network Address.
- Select Enabled under SSL for HTTP and click OK. (See Figures 6, 7, and 9.)
- Restart the agent.

STEP 4 TEST THE SSL LINK

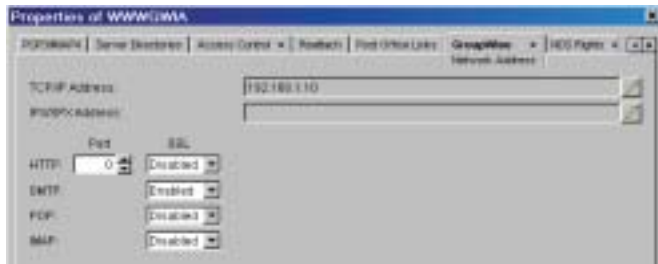
Test the new functionality by pointing a browser to the proper IP address and port of the agent's HTTP listener. If everything is working properly, the browser should automatically establish an SSL connection. Note the https:\\ URL in the address line in Figure 11.

SECURING USERS' WEBACCESS SESSIONS WITH SSL

GroupWise WebAccess User sessions aren't secured in any way by default. So of all of your efforts to secure your system, securing users' WebAccess sessions is arguably the most important. Enabling SSL for WebAccess User sessions is a simple matter of

FIGURE 9

When enabling SSL on the GroupWise Internet Agent other Internet hosts that support SSL will communicate via SSL/TLS to the GWIA.



enabling SSL on the Web server. This section explains how to do this on either an Apache Web server or a Novonyx Web server.

This section assumes that you already have an Apache Web server on a NetWare server running GroupWise 6.5 WebAccess. Now you want to enable SSL with your Apache Web server:

STEP 1 ADD A REFERENCE TO SSL CERTIFICATE DNS TO THE APACHE WEB SERVER

- Locate the *.CONF file for your Apache Web server. The *.CONF file is generally either GWAPACHE.CONF, ADMINSEV.CONF or HTTPD.CONF. The files are typically in the SYS:APACHE\CONF directory.
- Add the following four lines to either the top or the bottom of your *.CONF file:
LoadModule tls_module modules/mod_tls.nlm
<IfModule mod_tls.c>
SecureListen 192.68.1.10:443 "SSL CertificateDNS"
</IfModule>

NOTE: Change the IP address "192.68.1.10" to the TCP/IP address of the NetWare server running your Apache Web server.

NOTE: Although the name of the Key Material Object in eDirectory that you are referencing in this *.CONF file is SSL CertificateDNS - <server name> don't include the server name as part of the name because it is assumed.

NOTE: The words "SSL CertificateDNS" should be in quotes so it should read just as it shows in the example above.

Spam

in the office



YES
TESTED &
APPROVED

Novell

www.gwava.com

Your GroupWise® system is stuffed with junk email—and the spam just keeps on coming. Clear it out with GWAVA, the only email filter specifically designed for GroupWise. GWAVA protects your entire GroupWise system with:

- Anti-Virus protection
- Anti-Spam protection
- Email content & filtering

Learn how GWAVA can protect your GroupWise system from the inside out. Call 1-866-GO GWAVA, or e-mail us at: info@gwava.com.

FIGURE 10

With Telnet you can confirm that the GWIA is supporting SSL/TLS. The SMTP extension for SSL/TLS support is called STARTTLS on line seven of this screenshot.



FIGURE 11

GroupWise Agent HTTP monitoring sessions can even be secured via SSL.



- Bring the Apache Web server down and then back up.

The Web server should indicate that it is listening at port 443 (SSL for Web servers). (See Figure 12.)

STEP 1 ADD A REFERENCE TO SSL CERTIFICATE DNS TO NOVONYX WEB SERVER

- Locate the MAGNUS.CONF file for your Novonyx Web server. The *.CONF file is typically located in the SYS:NOVONYX\SUITESPOT\HTTPS-<File Server Name>\CONFIG directory.
- Add the following two lines to the bottom of the MAGNUS.CONF file:
Certfile SSL CertificateDNS
Keyfile SSL CertificateDNS

NOTE: Although the name of the Key Material Object in eDirectory that you are referencing in this *.CONF file is SSL CertificateDNS - <server name> don't include the server name as part of the name because it is assumed.

- Bring the Novonyx Web server down and then back up.

STEP 2 TEST FOR SSL SUPPORT

- In your Web browser, access the Web server with the following syntax:
https://<YourWebserver's DNS Address>

GETTING A CERTIFICATE SIGNED BY AN EXTERNAL CA

Using the SSL Certificate DNS Key Material Object (KMO) has potential drawbacks. The public key is signed by your internal Certificate Authority which is not preregistered in any Web browser. Web browsers recognize a handful of well-known Certificate Authorities (e.g., Verisign). Because of this, when your Web server serves up a public certificate that is not registered with a well-known Certificate Authority the Web browser prompts the user if they want to accept the certificate. For PC-based browsers, this is a mere annoyance. For users on wireless devices, particularly hosted or proxy-server-assisted devices such as Palm Web clipping devices, it's the kiss of death. The wireless devices and/or proxy server will not consume the certificate and users won't be able to access GroupWise WebAccess. Additionally, wildcard certificates (certificates that are registered to a domain, but not to a specific server), often don't work with wireless devices.

Take these steps to address this issue at a high level:

- Create a new custom Key Material Object (KMO) in eDirectory, e.g., an object called VERISIGN-SSL.
- Generate a Certificate Signing Request (CSR) from the KMO object.
- Submit the CSR to an external CA such as Verisign.
- Import the Certificate that the external CA sends back to you into eDirectory.
- Reference the custom KMO in the *.CONF file of your Web server. For example, on Apache the syntax would be:
LoadModule tls_module modules/mod_tls.nlm
<IfModule mod_tls.c>
SecureListen 192.68.1.10:443 "VERISIGN-SSL"
</IfModule>

FIGURE 12

An Apache Web server listening on port 443 with SSL.

```
Loading module MOD_JK.NLM
Loading module MOD_TLS.NLM
Apache/1.3.26 (NETWARE) mod_jk/1.1.0 running...
Listening on port(s): 443 80
Loaded dynamic module mod_jk.c
Loaded dynamic module mod_tls.c
```

For detailed instructions on how to get a certificate in your tree signed by an external certificate authority, see *Securing a Web Server on NetWare* online at <http://www.novell.com/connection/magazine/2002/10/secure.pdf>.

SECURING GROUPWISE MONITOR APPLICATION CONNECTIONS WITH SSL

Securing the GroupWise Monitor Application is identical to the previous section, *Securing Users' WebAccess Sessions With SSL*, because the GroupWise Monitor Application is also hosted on a Web server. Simply follow the same steps mentioned in that section. Note that the GroupWise Monitor Application can also run on wireless devices.

SECURING GROUPWISE MESSENGER (INSTANT MESSENGER) SESSIONS WITH SSL

The GroupWise Messaging Agent and the GroupWise Messenger Client can be configured to speak to one another via SSL:

STEP 1 ADD THE CERTIFICATE TO THE AGENT & ENABLE SSL

Perform these steps only if this agent has not yet had a certificate associated with it.

- Place your minted certificate on the server that is running the Novell Messenger/GroupWise Messenger Messaging Agent.
- From ConsoleOne, bring up the properties of the Messaging Agent Object.
- From the Agent tab select Security.
- Fill in the fields, specifying the certificate file and key file. (See Figure 13.)

XXX
in the office



www.gwava.com

Is your GroupWise® system starting to look like a seedy magazine stand? Clean it up with GWAVA, the only email filter specifically designed for GroupWise. GWAVA protects your entire GroupWise system with:

- Anti-Virus protection
- Anti-Spam protection
- Email content & filtering

Learn how GWAVA can protect your GroupWise system from the inside out.

Call 1-866-GO GWAVA,
or e-mail us at:
info@gwava.com.

FIGURE 13

The GroupWise Messenger Agent (Instant Messaging) is configured with a certificate and a key, just like the other GroupWise agents.

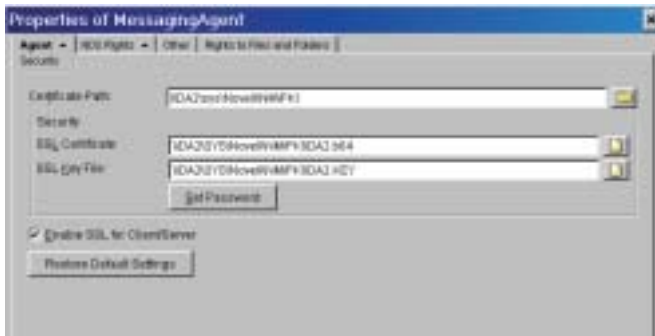
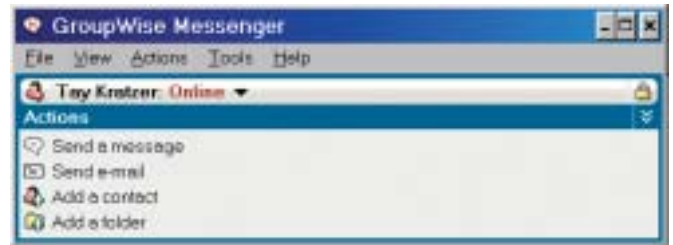


FIGURE 14

When the GroupWise Messenger Agent and the GroupWise Messenger Client are communicating via SSL, the GroupWise Messenger client shows a padlock icon in the upper right-hand corner of the interface.



- Click on Set Password and enter the case-sensitive password.
- Mark the Enable SSL for Client/Server check box.
- Exit out of the dialog.

STEP 2 TEST FOR SSL SUPPORT

- Log in to the GroupWise Messenger client.
- Note the padlock icon on the GroupWise Messenger client screen. (See Figure 14.)

SUMMARY

You have now secured your entire GroupWise system via SSL, giving you the most secure messaging solution possible. You can rest assured that all of your electronic transmissions will be protected by today's leading encryption technology. Not only will you breathe easier every time you read about the latest security breaches with other messaging solutions, but you will also be able to meet any security requirements thrown your way. Whether it be HIPAA, banking regulations, or any other security concerns, you can now implement, document and prove that you have the tightest security available. So go ahead and throw away that pen and paper, because all you need is GroupWise 6.5!

We've described how to secure all message transmissions. If you use LDAP authentication in GroupWise or GroupWise Messenger, you'll want to secure your authentication data transmission also. For more information on secure LDAP authentication, see Chapter 27 of the *Novell Press GroupWise 6.5 Administrator's Guide* online at www.novellpress.com. **N**

The SSL Architecture

Whenever anyone talks about data security, you will invariably hear the term Secure Sockets Layer (SSL). At its core, SSL uses long numerical strings to encode raw data into data streams that can only be decoded by the intended recipient. The actual numerical strings used to encrypt and decrypt the data are called *keys*.

SSL supports both symmetric-key encryption and public-key encryption. With symmetric-key encryption the server and client agree on a key which is used for both encrypting and decrypting. This method works well in many cases and is faster than public-key encryption, but cannot provide the authentication strength of public-key encryption. An example of this in a GroupWise environment is the encryption between the WebAccess Application (Web server servlet) and the WebAccess Agent (GWINTER). The symmetric key is kept within the COMMGR.CFG file used by both the WebAccess Application and the WebAccess Agent.

Public-Key Infrastructure (PKI) is the technology which covers public-key cryptography. When dealing with PKI you always generate keys in pairs resulting in a public key and a private key. As the name implies, your public key can be shared with anyone while your private key must always remain secret. In order to send an encrypted message, the message is encrypted with the intended recipient's public key. The recipient would then decrypt the message with the corresponding private key.

Key pairs are generated by complex mathematical equations. The key pairs are generated in a way that only the corresponding matching key can be used to decrypt a message. Encrypted messages can only be decrypted with the matching private key, and not even the key used for encrypting the message can decrypt it.