MYOB Greentree

Security Recommendations

March 2022

Contents

Introduction 1

User Accounts 2

Configure Account Settings 2

Windows Accounts 4

Use a Service Account 4

Windows Apps 6

Block Fat Client Applications 6

Restrict Thin Client Applications 7

Encrypt Thin Client Communication 8

Use OAuth 2 for Exchange Web Services 10

Web Apps 12

Configure eModules 12

Secure API 13

Web Server 14

Add Response Headers 15

Block Short Filename Disclosure 17

Configure Default Error Pages 18

Require HTTPS 21

Restrict Response Headers 22

Database 25

Encrypt the Database 25

Configure Attachment Security 27

Introduction

MYOB takes security seriously. We want to provide MYOB Greentree Partners with the best advice for securing Greentree sites.

This document describes practices that will increase the security of Greentree systems and the data they hold. It identifies the intention of each practice, describes how it increases security, signals implications of adopting the practice and gives detailed instructions.

Although great care has been taken in the preparation of these recommendations, implementation, enforcement and verification remain the responsibility of Partners.

We expect to expand and to refine this material and to publish new versions from time to time.

Version	Summary	Publication Date
1	Initial release	July 2019
2	Added database encryption and application restrictions	January 2020
3	Added account settings, web server recommendations, attachment security and OAuth2	February 2021
4	Extended web server recommendations	November 2021
5	Extended OAuth2 recommendations for eDocs	December 2021
6	Extended OAuth2 recommendations for email sending	March 2022

In this version, the recommendations were updated to include the facility released in Greentree 2022.1 for using OAuth2 for authentication when sending email using Exchange Web Services.

User Accounts

This section contains recommendations for managing user access to Greentree.

Configure Account Settings

Greentree allows site administrators to enforce strong password management rules on Greentree user accounts. This is highly configurable and easy to apply.

Recommendations

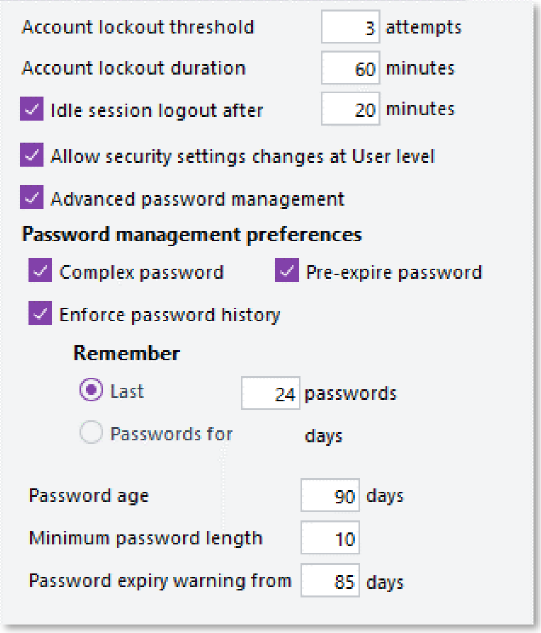
1. Force accounts to be locked out after successive attempts to sign in with an incorrect password.
2. Force sessions to be closed after being idle.
3. Turn on the Advanced Password Management feature and configure rules for password complexity, length, expiry, lockouts and re-use.
4. Configure maximum password attempts for eHR, eTimesheet and eService users.

The Idle Session Logout feature applies to Windows client and Browser sessions, but not eModules. Account locking and Advanced Password Management applies to Windows client and Browser sessions. From Greentree release 2021.1 password complexity and history rules also apply to eModule users – contacts, organisations, JC employees and HR employees – provided Advanced Password Management is turned on.

Instructions

1. Sign in to Greentree as an administrator and locate the **Account settings** tab of the General System Preferences form.
2. Set the **Account lockout** threshold (between 2 and 9 attempts) and duration (up to 1 day).
3. Turn on **Idle session logout** and set the idle period.

4. Turn on **Advanced password management** and apply the recommended settings shown here:



Account lockout threshold 3 attempts

Account lockout duration 60 minutes

☑ Idle session logout after 20 minutes

☑ Allow security settings changes at User level

☑ Advanced password management

Password management preferences

☑ Complex password ☑ Pre-expire password

☑ Enforce password history

Remember

● Last 24 passwords

○ Passwords for days

Password age 90 days

Minimum password length 10

Password expiry warning from 85 days

The full detail of settings such as Complex Password is described in the Greentree online help.

5. Locate the **eModule Control** form. Depending which modules are licensed and enabled, this can be found using these menu paths:
   System > eTimeSheets > Setup
   SCM > System > eRequisitions > Module Control
   CRM > System > eService > Module Control
   CRM > System > eCRM.
6. Set Max Login Attempts to 3.
7. Locate the **eHR Module Control** form using this menu path:
   HR > System > eHR > Setup > Module Control.
8. Set Max Login Attempts to 3.

Windows Accounts

This section deals with the Windows accounts under which Greentree services run.

Use a Service Account

The principle of least privilege is that a program should have the minimum access to resources needed for its purpose. This should be applied to the Jade programs that are the basis for the Greentree system.

The standard production configuration for a Jade site is to run the Jade database server (*jadrap*) and application servers (*jadapp*) as Windows services. This allows the services to be started and shut down automatically.

By default, Windows services run under the default Local System account. This is a pre-defined account with extensive privileges. It is not recommended for use with Greentree services.

Recommendations

Use a service account created and configured specifically for Greentree services. The service account should be configured so that the password never expires. Once you have created the new service account, it will need full control of the Jade installation directory.

If your database server and application servers run on different machines, you'll need to create a service account on each machine and configure each account with permissions on its corresponding machine. The site's IT infrastructure may require the accounts to be whitelisted in firewall rules to allow communication.

If the site runs Microsoft's Active Directory service, you should consider creating a Managed Service Account. Supporting material can be found here https://blogs.technet.microsoft.com/askds/2009/09/10/managed-service-accounts-understanding-implementing-best-practices-and-troubleshooting/.

Instructions

Locate *Local Users and Groups* on the Computer Management application.

Create the service account with a meaningful name such as *GreentreeServiceAcc*.

Turn on setting *Password never expires*.

Deny this user permissions to log on to Remote Desktop Session Host server.

On Dial-in, Network Access Permission, select *Deny access*.

Grant the account full control of the Jade installation directory e.g. C:\Greentree and and all its subdirectories and files by:

- Opening Windows File Explorer.
- Locating the Jade installation directory.
- Choosing **Properties** from the pop-up menu.
- Clicking on the **Security** tab.

- Clicking on the **Edit** button.
- Adding the account and granting it **Full Control**.

Modify the Windows service to use the account by:

- Opening Windows services.
- Locating the service.
- Choosing **Properties**.
- Clicking the **Log On** tab.
- Choosing the **This Account** option.
- Clicking the Browse button,
- Searching for and selecting the account
- Entering the account's password and clicking **OK**.

Confirm that services start and stop correctly.

Windows Apps

This section deals with the security of data in transit for Greentree Windows applications.

Block Fat Client Applications

Jade provides two mechanisms for Windows client connections. A Jade **thin client** is a presentation client. It is a light-weight executable that runs on a user's Windows computer. It handles the graphical user interface and communicates with the remote Jade application server.

A Jade **fat client** is a Windows executable that manages a Jade database node. The node holds persistent and transient data at rest. This is necessary for some Greentree services that run on Windows servers; this can be the same machine where the primary database resides or another machine. System administrators can restrict access to these servers but cannot easily restrict access to Jade fat clients running in other locations. There should be no need for users to employ fat client access to a production Greentree site.

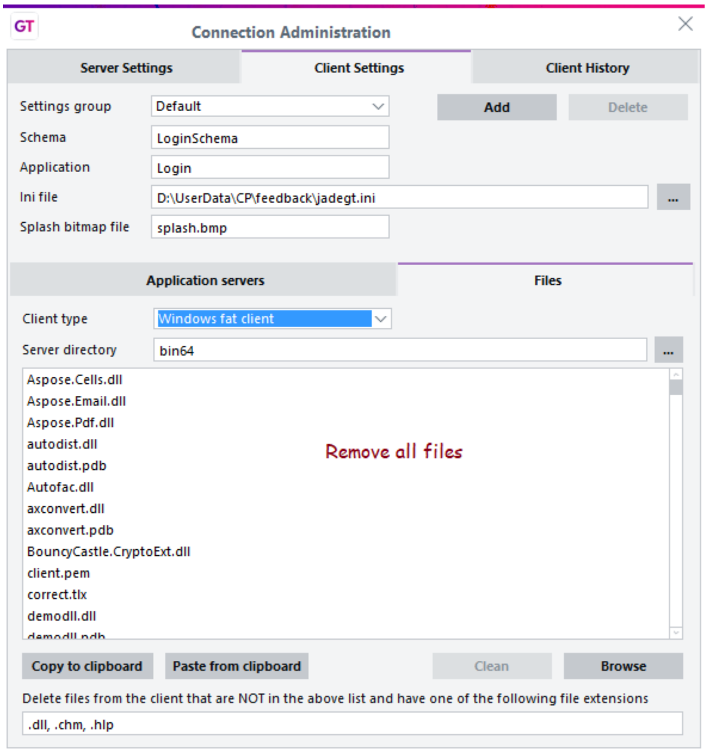
Recommendations

Do not configure fat client user access to Greentree.

Block existing fat client user access via Connection Manager.

Instructions

Run the Connection Admin application. Click the **Client Settings** tab, and the **Files** tab beneath it. For each Settings group, select the client type **Windows fat client**. Clear the list for this client type under all settings groups by selecting all (Ctrl+A) and using the Delete key.



Settings group: Default

Schema: LoginSchema

Application: Login

Ini file: D:\UserData\CP\feedback\jadegt.ini

Splash bitmap file: splash.bmp

Client type: Windows fat client

Server directory: bin64

Aspose.Cells.dll
Aspose.Email.dll
Aspose.Pdf.dll
autodist.dll
autodist.pdb
Autofac.dll
axconvert.dll
axconvert.pdb
BouncyCastle.CryptoExt.dll
client.pem
correct.tlx
demodll.dll
demodll.pdb

Remove all files

Delete files from the client that are NOT in the above list and have one of the following file extensions

.dll, .chm, .hlp

Restrict Thin Client Applications

The normal access to the Greentree Windows client is initiated using Greentree's Connection Manager executable. This will launch only those applications configured using Connection Administration. However, users with knowledge of an application server's address and port number can attempt to start other applications using the Jade executable in a local directory. Fortunately, Jade provides a way of enforcing a white-list of allowed applications.

Recommendation

Configure the INI file used by application servers to restrict applications that can be launched by thin client applications.

Instructions

Turn on the EnableAppRestrictions setting in the [JadeAppServer] section of the application server's INI file, and specify a whitelist of applications using AllowSchemaAndApp{n} settings as illustrated below:

```
[JadeAppServer]
EnableAppRestrictions=true
AllowSchemaAndApp1=LoginSchema,Login
AllowSchemaAndApp2=LoginSchema,LoginWithTimeoutWarning
```

Encrypt Thin Client Communication

The recommended form of user access to the Greentree Windows application is via Jade **thin client**. This is a *presentation* client, a light-weight executable that runs on a user's Windows computer. It handles the local graphical user interface and communicates with the remote Jade application server via TCP/IP. This communication is vulnerable to attacks, but the risk can be mitigated using strong encryption.

Recommendation

In our recommended solution the client authenticates the application server, and communication is encrypted using SSL/TSL but the application server does not authenticate the presentation client. This requires each site obtaining, applying and updating an SSL certificate on the application server.

For more information and more advanced solutions see Jade Smart Thin Client Security.

Instructions

Obtain a valid SSL certificate along with the private key. Store them in a secure folder on the server. For example, store certificate.pem and privatekey.pem in C:\GTCertificates. The certificates must be structurally correct and be signed by a valid Certificate Authority. Checks are not performed on the validity of the certificate date range nor that the connection is to whom the certificate says it should be. The certificate files must be in PEM-encoded format, and they cannot require passphrases.

Add the following to the JadeAppServer section of the application server's INI file. Set SSLSecurePort to the port of the app server.

```
[JadeAppServer]
RPCEncryptionEnabled=true
RPCEncryptionHookDLL=SSL_TLS
SSLPrivateKeyFile=C:\GTCertificates\privatekey.pem
SSLCertificateFile=C:\GTCertificates\certificate.pem
SSLSecurePort=50011
```

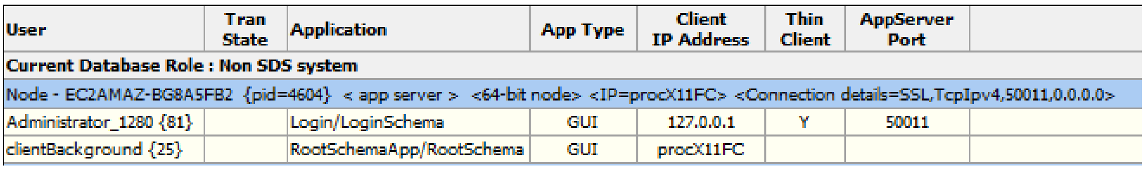
Add the following to the JadeThinClient section of the client INI file specified in Connection Manager:

```
[JadeThinClient]
RPCEncryptionEnabled=true
RPCEncryptionHookDLL=SSL_TLS
SSLSecurePort=50011
```

For internal testing, use OpenSSL to generate a self-signed certificate with the command below. Note that MYOB does not recommend using self-signed certificates on production systems.

```
openssl req -newkey rsa:2048 -new -nodes -x509 -days 365 -keyout
privatekey.pem -out certificate.pem
```

To confirm the application server has started correctly, check that it appears with details=SSL in the Jade Monitor:



User	Tran State	Application	App Type	Client IP Address	Thin Client	AppServer Port	
Current Database Role : Non SDS system							
Node - EC2AMAZ-BG8A5FB2 {pid=4604} < app server > <64-bit node> <IP=procX11FC> <Connection details=SSL,TcpIpv4,50011,0.0.0.0>							
Administrator_1280 {81}		Login/LoginSchema	GUI	127.0.0.1	Y	50011	
clientBackground {25}		RootSchemaApp/RootSchema	GUI	procX11FC			

When a thin client has connected to the app server, the lines like these should appear on the jommsg log:

```
2019/07/26 07:11:41.499 02708-4da0 JadApp: Client Tcp address=11.11.11.11 port=53561, App Server Tcp address=22.22.22.22 port=50011, protocol=2

2019/07/26 07:11:41.612 02708-4da0 jomsec: Current cipher DHE-RSA-AES256-GCM-SHA384, strength 256 bits.
```

Use OAuth 2 for Exchange Web Services

Three Greentree functions retrieve documents via email using Microsoft Exchange Web Services: CRM Email Filing, eDocs and eXchange EDI. In addition, Greentree can send mail using Exchange Web Services. To date, Exchange has supported two forms of authentication: basic authentication and OAuth 2, which is more secure.

With basic authentication, a Greentree user provides the username and password required to access an email account, and those credentials are stored in the database until needed by the email retrieval function.

With OAuth 2, Greentree directs the user to a Microsoft web page that captures the username and password for the email account. The Microsoft web page then passes a token back to Greentree, which is stored in the database.

In 2020, Microsoft announced that basic authentication would be phased out. The initial change only affected publicly accessible Exchange Servers, but privately managed servers could continue to use basic authentication.

Until version 2020.5, Greentree supported only basic authentication. From version 2021.1, Greentree also supports OAuth 2 for CRM Email Filing and eXchange EDI. From version 2021.4, Greentree also supports OAuth2 for eDocs. From version 2022.1, Greentree also supports OAuth2 for email sending. Both authentication methods store credentials in the database using reversible encryption, and decrypt them on the fly when connecting to mail servers. Greater security for data at rest can be achieved by encrypting the database.

Recommendations

1. If the server providing Exchange Web Services has OAuth 2 enabled, change the credentials in your Greentree system to use OAuth 2 instead of basic authentication.
2. If the server does not have OAuth 2 enabled, make it so!
3. Use Greentree’s **Token Management** function to ensure OAuth 2 credentials are kept usable.

Instructions

Locate basic credentials in the following places:

- For each company that uses CRM Email Filing, open the CRM Module Control form (CRM > System > Module Control), locate the Exchange Account settings on the Main tab.
- For each row of the table on the Main tab of the eDocs Module Control form (System > eDocs > Module Control) that has a type of email.
- For each EDI Profile that retrieves documents by email, on the Source by Email tab of the eXchange EDI Profiles form.
- For each EDI Email Import Task (System > eXchange EDI > Tools > Email Fetch Tasks).
- On the Emailing Preferences tab of the General System Preferences form (System > System Setup > General System Preferences).
- On the Company Maintenance form (System > System Setup > Company Maintenance), open the Emailing Preferences tab and step through each company; identify companies where the Use Company settings is turned on.

In each case, click the **Credentials** or **Change** button, change the **Authentication method** from “Basic” to “OAuth2”, save and click **Log in**. The user is redirected to a Microsoft sign-in page, where they can log in with their OAuth 2 details. Once the user is successfully authenticated, their OAuth login token is stored in the Greentree system. You can use the Token Management window (System > Utilities > Token Management) to check that users’ OAuth tokens are kept up to date, and set email reminders for when the tokens are due to expire.

Web Apps

This section deals with the security of data in transit for Greentree web applications. It contains recommendations that are specific to eModules and the Greentree API.

Configure eModules

The eModule applications offer control over the facility where a user can request a password to be reset with a new password being sent by email. They also allow demonstration facilities based on credentials stored in INI files.

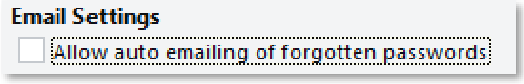
Recommendations

- Turn off the facility for resetting and emailing passwords in Greentree.
- Remove demonstration facility credentials from configuration files on the web server.

Instructions

To turn off the password reset facility:

1. Open the **eModule Control** form by selecting **CRM > System > eCRM > Module Control** or **CRM > System > eService > Module Control**.
2. On the **Main** tab, deselect the **Allow auto emailing of forgotten passwords** setting.



Email Settings

□ Allow auto emailing of forgotten passwords

To remove the demonstration facility:

1. Locate the eGreentree.ini files in each of the eModules directories on the web server. For example, C:\inetpub\wwwroot\Greentree\eApprovals\configuration\eGreentree.ini
2. In the **[Login]** section of the INI file, remove the lines that begin with:

- DemoCompanyCode
- DemoCustomerCode
- DemoCustomerPassword
- DemoEmployeeCode
- DemoEmployeePassword
- DemoExternalCode
- DemoExternalPassword
- DemoUserCode
- DemoUserPassword
- ShowDemoButton.

Secure API

The Greentree API can be secured by a technique called *reverse proxy*: communication between clients and the web server uses HTTPS and requires a certificate on the server, but communication between IIS and the API application running on the Jade database server uses TCP/IP.

Recommendation

Set up a reverse proxy using Microsoft IIS Manager. In preparation, devise a URL that is short and clear, like: https://hostname/greentree/api/01/APInvoice. Your URL should include:

- A unique identifier for the system. This can be as simple as “greentree”.
- The word “api”, which distinguishes this from other applications on the same system.

Implement this by configuring a pattern on the inbound rule. For example, greentree/api/* as explained in the instructions.

Instructions

See the Greentree API documentation at:

https://jubbly.atlassian.net/wiki/spaces/GTAPI/pages/14352392/Achieving+an+SSL+connection+by+configuring+IIS+as+a+Reverse+Proxy

Web Server

Greentree uses Microsoft's Internet Information Services (IIS) for its API, Browser, eModules, WebView and Webstore applications. This section contains some general recommendations for securing IIS.

The recommendations are current as of September 2021. They are not intended to be a definitive guide. Responsibility for securing web servers lies with sites, and the requirements change over time. Sites should set up and secure web servers following standard practice. Operating system and IIS security updates should be applied.

Recommendations for securing IIS are outlined on the Microsoft website and in the IIS 8 Server Hardening Handbook.

This guide gives instructions for configuring IIS using the Internet Information Services Manager application. Experts can achieve the same results by editing XML configuration files, by running the AppCmd command line tool or by using WMI scripts.

Add Response Headers

Adding response headers helps protect users from malicious attacks like content sniffing, cross-site scripting (XSS) and framing.

Content sniffing

Content sniffing is the client-side activity of inspecting the content of data provided by a server to learn that data's format. This can be part of tricking a browser into executing a script that is disguised as another file type.

Adding the **X-Content-Type-Options** response header can help protect users from malicious content.

Cross-site scripting (XSS)

Cross-site scripting is the injection of client-side scripts into web pages. Reflected or non-persistent cross-site scripting involves scripts submitted by a client and reflected by a server.

Adding the **X-XSS-Protection** response header can reduce the risks. This header directs the behaviour of the browser when it detects content that could be used for reflected cross-site scripting attacks. With the *block* setting, a browser stops loading pages in this situation.

Framing

Framing is the hijacking of a web application so that it runs in an external site.

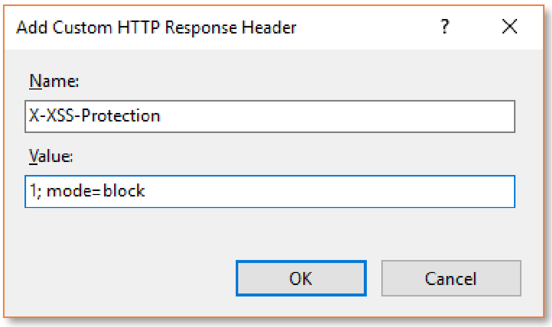
Adding the **X-Frame-Options** response header can prevent this. Since this header blocks browser pop-up windows, which are used extensively by eModules, it should not be added for sites running eModules.

Recommendation

Configure HTTP headers that are added to the web server's responses for all web sites.

Instructions

1. Open IIS Manager.
2. In the **Connections** panel on the left, select the server.
3. In the **IIS** section, double-click **HTTP Response Headers**.
4. In the **Actions** panel on the right, click **Add…** to add headers.
5. To block content sniffing, add a response header with the name **X-Content-Type-Options** and the value **nosniff**.
6. To block cross-site scripting, add a response header with the name **X-XSS-Protection** and the value **1; mode=block**.



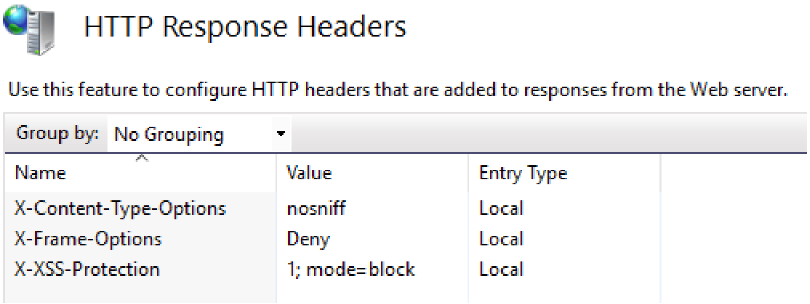
Add Custom HTTP Response Header

Name: X-XSS-Protection

Value: 1; mode=block

7. On sites that do not run eModules, to prevent framing, add a header with name **X-Frame-Options** and the value **Deny**.

   After adding the headers, they're defined at the server level and inherited by all applications:



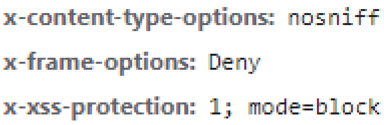
HTTP Response Headers

Use this feature to configure HTTP headers that are added to responses from the Web server.

Group by: No Grouping

Name	Value	Entry Type
X-Content-Type-Options	nosniff	Local
X-Frame-Options	Deny	Local
X-XSS-Protection	1; mode=block	Local

8. Confirm this is effective by running an application, opening the browser's developer tools and inspecting the response headers:



```
x-content-type-options: nosniff
x-frame-options: Deny
x-xss-protection: 1; mode=block
```

Block Short Filename Disclosure

If short (8.3) file names are present or can be created, file names on the web server can be accidentally disclosed.

Recommendations

- Block the creation of short file names for new files. This does not remove short names for existing files.
- Configure IIS to block requests for URLs or query strings that containing tilde ~ characters.

Instructions

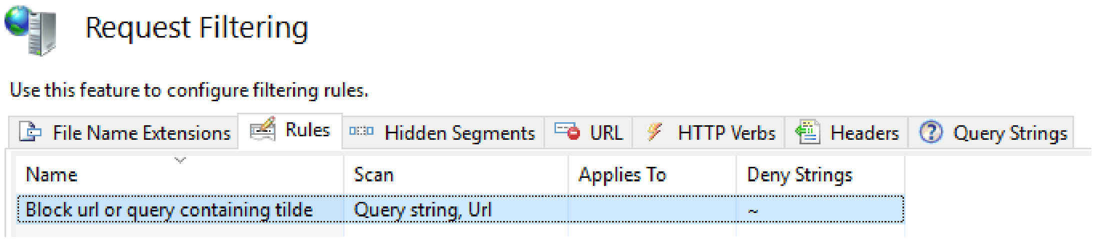
To disable creation of short file names:

1. On the web server, open a command window as an administrator.
2. Type this command: `fsutil behavior set disable8dot3 1`, then press **Enter**.

Note: To confirm the change, create a file with a long name, run dir /x and check that no short name is created.

To block requests containing tilde characters:

1. Open IIS Manager.
2. In the **Connections** panel on the left, select the server.
3. In the **IIS** section, double-click **Request Filtering**.
4. Click the **Rules** tab.
5. In the **Actions** panel on the right, click **Add Filtering Rule…**.
6. Select the **Scan url** and **Scan query string** options.
7. In the **Deny Strings** section, enter a tilde (~).



Request Filtering

Use this feature to configure filtering rules.

Name	Scan	Applies To	Deny Strings
Block url or query containing tilde	Query string, Url		~

Configure Default Error Pages

The default error pages that IIS shows can include details useful for hackers. You can reduce this risk by configuring custom error pages that disclose less information.

Resources for the Greentree browser include custom pages for common HTTP errors. You can also use these pages for other applications.

Recommendation

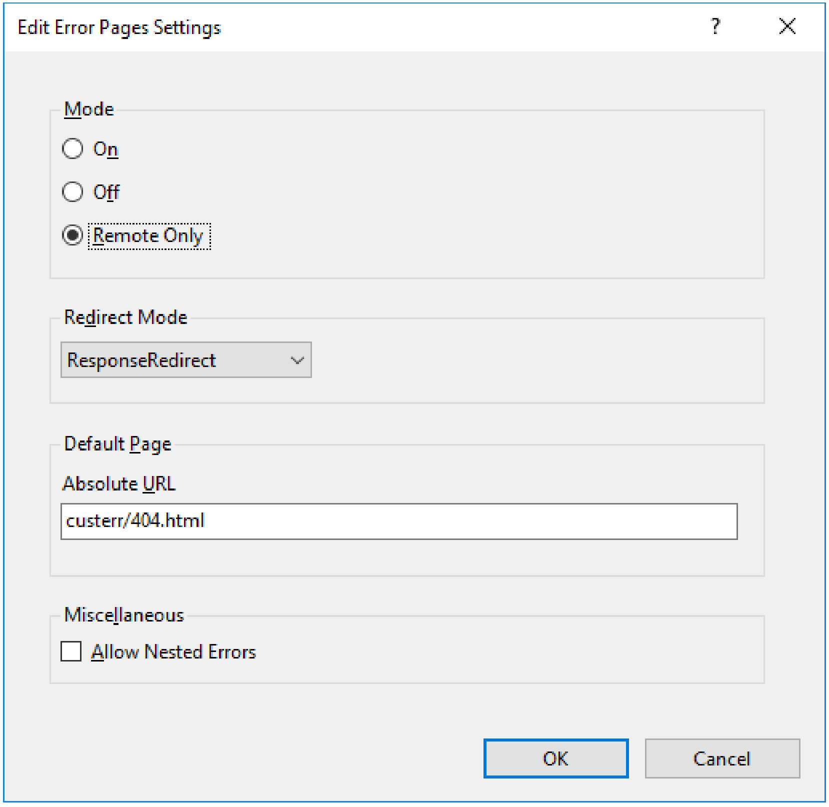
Configure IIS to use a single set of custom error pages for all Greentree web applications. If your IIS setup places all applications under a single web site, you can perform the setup for the web site and have the applications inherit the configuration.

Instructions

Note: If the **HTTP Errors** feature is not available in IIS Manager, turn it on in Server Manager, under **Web Server (IIS)** > **Web Server** > **Common HTTP Features**.

1. Open IIS Manager.
2. In the **Connections** panel on the left, locate your Greentree web applications. These might include the Greentree browser, six eModules (eApprovals, eCRM, eHR, eRequisitions, eService and eTimesheets) and WebView.
3. If the applications have a common parent node that has only Greentree applications beneath it, you can set up on that parent node. Otherwise, to avoid showing Greentree custom error pages for non-Greentree applications, you need to repeat the setup for each Greentree application.
4. Identify the node’s physical directory.
5. Open the physical directory in file explorer.
6. Create a subdirectory named **custerr**.
7. In the **custerr** subdirectory, copy the six Greentree browser error pages from browser\resources\custom_errors\*.html
8. Go back to IIS Manager.
9. Select the node in the left panel, locate the .NET Error Pages item in the middle panel and open it.
10. In the **Actions** panel on the right, click **Edit Feature Settings**.
11. In the **Edit Error Pages Settings** window, complete the following sections:
    - In the **Mode** section, select **Remote Only**.
    - In the **Redirect Mode** section, select **ResponseRedirect**.

- In the **Default Page** section, enter the URL `custerr/404.html`.



Edit Error Pages Settings

Mode

○ On

○ Off

● Remote Only

Redirect Mode: ResponseRedirect

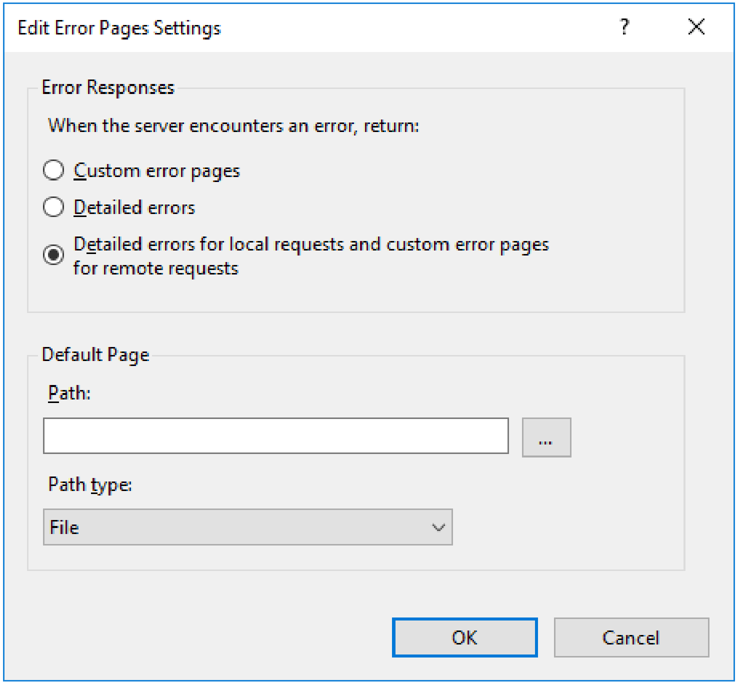
Default Page

Absolute URL: custerr/404.html

Miscellaneous

□ Allow Nested Errors

12. Click **OK** to close the window.
13. In the left panel, reselect the node.
14. In the middle panel, open the **Error Pages** item.
15. In the **Actions** panel on the right, click **Edit Feature Settings**.
16. In the **Edit Error Pages Settings** section, complete the following sections:
    - In the **Error Responses** section, select **Detailed errors for local requests and custom error pages for remote requests** option.
    - From the **Path Type** dropdown, select **File**.



Edit Error Pages Settings

Error Responses

When the server encounters an error, return:

○ Custom error pages

○ Detailed errors

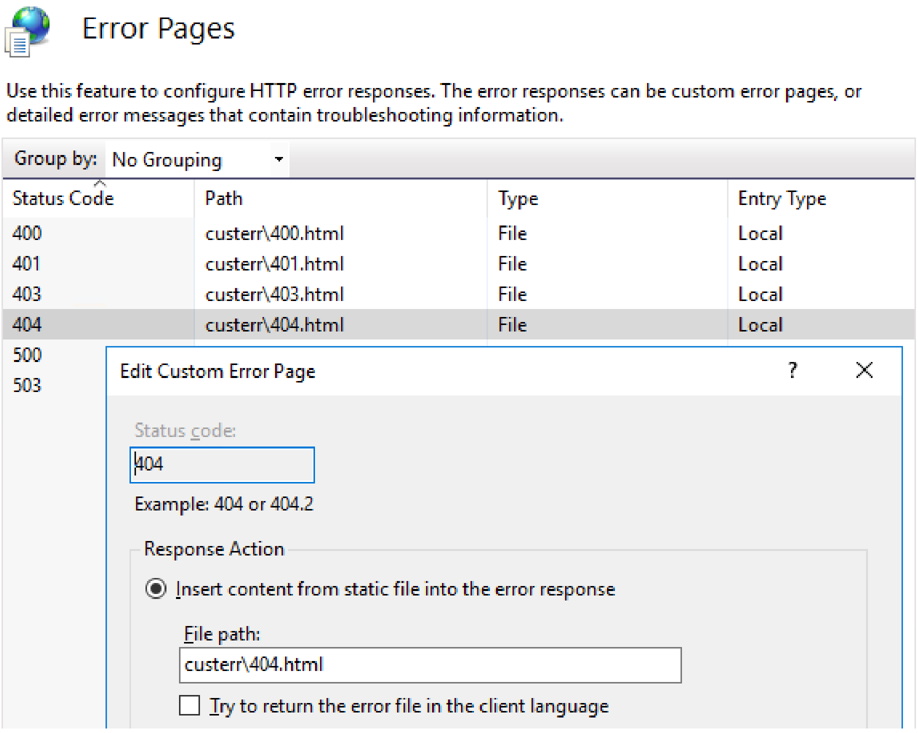
● Detailed errors for local requests and custom error pages for remote requests

Default Page

Path:

Path type: File

17. Configure an entry for the six pages as illustrated below. If there is an *inherited* page for a matching status code, delete it and add a new one; this will show the entry type as *Local* on this display, and *Inherited* on the equivalent display for child nodes.



Error Pages

Use this feature to configure HTTP error responses. The error responses can be custom error pages, or detailed error messages that contain troubleshooting information.

Group by: No Grouping

Status Code	Path	Type	Entry Type
400	custerr\400.html	File	Local
401	custerr\401.html	File	Local
403	custerr\403.html	File	Local
404	custerr\404.html	File	Local
500			
503			

Edit Custom Error Page

Status code: 404

Example: 404 or 404.2

Response Action

● Insert content from static file into the error response

File path: custerr\404.html

□ Try to return the error file in the client language

18. Remove any local overrides in child applications.
    - Go through each node beneath the node you have been configuring, open the **Error Pages** item and confirm that you see only inherited error pages, and that the **Feature Settings** show the inherited setting for **Error Responses** (**Detailed errors for local requests…**).
19. If you don't see the inherited error pages in child applications, you need to directly change the web.config file in the corresponding directory using a text editor.
    - Locate the XML element at configuration/system.webServer/httpError. If this contains a **<clear />** tag, remove it, save the file then reselect the node in IIS Manager. Check the inherited **Error Page** settings have restored.

To test your setup:

1. Run a browser on a remote machine and enter an invalid URL e.g. http://hostipaddress/greentree/greentree-desktop/doesnotexist. This should display the Greentree 404.html page.
2. Enter a URL for a virtual directory and confirm you get the Greentree 403.html page e.g. http://hostipaddress/greentree.

Hints:

- If you perform the checks using a browser on the server, you will get detailed ASP.NET and IIS error pages.
- IIS stores the configuration for each node in a web.config file in the corresponding physical directory. To restore an inherited **Error Page** item that you have deleted, edit this file.

Require HTTPS

When communicating with IIS, Greentree client applications use a protocol configured on IIS. With HTTPS, this communication is encrypted. It involves the use of secure sockets layer (SSL) and makes client applications certain of the server's identity.

Recommendations

- Obtain a certificate for each site.
  - For production environments, you should obtain certificates from official providers known as certificate authorities.
  - For test and development environments, you can use self-signed certificates.
- Enforce the use of HTTPS for these applications in IIS.
- Provide redirection of HTTP requests to HTTPS.

Instructions

Follow the instructions on the Microsoft support site.

In summary, the steps are:

1. Obtain a certificate.
2. Install the certificate.
3. Configure the site's bindings for https and specify the certificate.
4. Turn on **Require SSL** in **SSL Settings** for the site, and confirm this is inherited by applications.
5. Test.

To configure redirection:

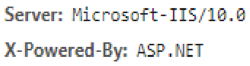
Note: If the **URL Rewrite** feature is not avaiable in IIS Manager, turn it on in **Server Manager**, under **Web Server (IIS)** / **Web Server** / **Common HTTP Features**.

1. Open IIS Manager.
2. In the **Connections** panel on the left, select the server.
3. In the middle panel, open **URL Rewrite**.
4. Add a blank rule, set name "Redirect HTTP to HTTPS", requested URL option Matches the Pattern, using Regular Expressions, pattern (.*), action type Redirect, redirect URL https://{HTTP_HOST}/{R:1}, turn on Append query string, redirect type Permanent (301).
5. Add a condition that will prevent the rule being applied to HTTPS requests: condition input {HTTPS}, check Matches the Pattern, pattern ^OFF$, turn on Ignore Case.

Test the redirection by entering a HTTP URL in a browser. Make sure that it redirects to the HTTPS equivalent.

Restrict Response Headers

By default, HTTP response headers reveal operating system, web server version numbers and settings that might help an attacker. Generally, it’s easy to blocking or remove this sort of information:



```
Server: Microsoft-IIS/10.0
X-Powered-By: ASP.NET
```

Recommendation

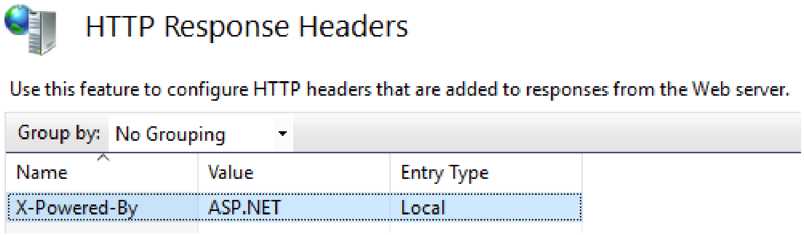
Follow the instructions given below for:

- Removing the **X-Powered-By** header.
- Removing the IIS version information shown in the server header.

Instructions

To remove the X-Powered-By header:

1. Open IIS Manager.
2. In the **Connections** panel on the left, select the server.
3. In the middle panel, open **HTTP Response Headers**.
4. Select the **X-Powered-By** header, then click **Remove** in the **Actions** panel on the right.



HTTP Response Headers

Use this feature to configure HTTP headers that are added to responses from the Web server.

Group by: No Grouping

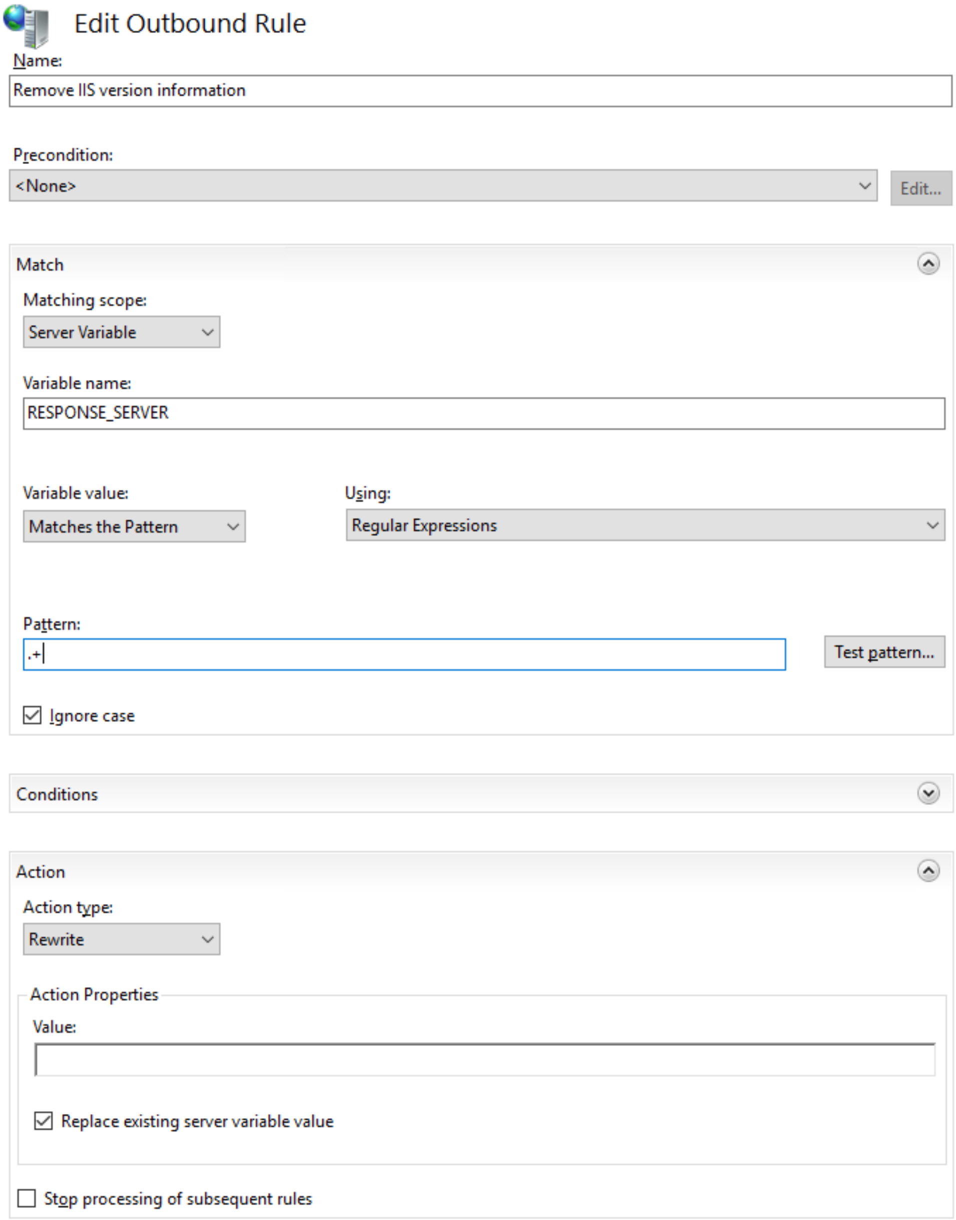
Name	Value	Entry Type
X-Powered-By	ASP.NET	Local

To remove the value shown in the server header:

Note: If URL Rewrite is not visible in IIS, download **URL Rewrite 2** from the Microsoft web site and install it.

1. Open IIS Manager.
2. In the **Connections** panel on the left, select the server.
3. In the middle panel, open **URL Rewrite**.
4. In the **Actions** panel on the right, click **View Server Variables**.
5. Add the variable, **RESPONSE_SERVER**.
6. Create an **Outbound Rule** with the following settings:
   - In the **Name** field, enter “Remove IIS version information”.
   - From the **Matching scope** dropdown, select **Server Variable**.
   - In the **Variable name** field, enter “RESPONSE_SERVER”.
   - From the **Variable value** dropdown, select **Matches the Pattern**.
   - From the **Using** dropdown, select **Regular Expressions**.
   - In the **Pattern** field, enter `.+`.
   - In the **Action** section, from the **Action type** dropdown, select **Rewrite**.

- In the **Action** section, leave the **Value** field empty.



Edit Outbound Rule

Name:
Remove IIS version information

Precondition:
<None>

Match

Matching scope:
Server Variable

Variable name:
RESPONSE_SERVER

Variable value:
Matches the Pattern

Using:
Regular Expressions

Pattern:
.+

☑ Ignore case

Conditions

Action

Action type:
Rewrite

Action Properties

Value:

☑ Replace existing server variable value

☐ Stop processing of subsequent rules

Note: This configuration is held in the applicationHost.config file typically in C:\Windows\System32\inetsrv\config.

To confirm the changes:

1. Open a browser’s development tools and view **Headers**.
2. Go to a web page.
3. Check that the **X-Powered-By** header is not present and that no value is shown for the server header.

Database

This section deals with the security of data at rest. It covers the database, database nodes, backups and the security of attachments.

Encrypt the Database

A Greentree system may hold a large amount of personally identifiable information and commercially sensitive data. Some security can be gained through process improvements such as reviewing user accounts. Some can be gained through configuration within Greentree such as applying Advanced Password Management. Some improvements can be gained only through highly technical means such as *whole database encryption*.

Jade's database encryption uses operating system facilities to secure database files. Windows holds keys tied to the Windows account under which the database services run. Data is encrypted when stored by Jade's Object Manager (jom) and decrypted when retrieved. Individual Greentree applications are unaffected and have no visibility of this process.

The default encryption algorithm is Advanced Encryption Standard (AES) with a 256-bit key. This is provided by the Microsoft software stack (module CNG) and meets the requirements of United States Federal Information Processing Standards (FIPS) 140-2 level 2 certification. (Jade Help)

The recommended use of this encryption with Greentree is:

- Encrypt the whole database rather than individual database (map) files.
- Apply the Default Security access check option.

The implications of using this form of protection include:

- **Secure handling of keys and passphrases is critical.**
  "If you lose the exported master key or its export file passphrase, you cannot move the database to another machine or restore the contents of the keystore if it is lost or corrupted. In this case, you will be unable to access or decrypt the contents of the database." (Jade Help)
- There may be minor performance degradation.
  Our testing has not yet identified the extent or characteristics of this.
- Backups are encrypted.

There are restrictions on encrypting Jade databases that may be significant for VADs:

- The system cannot use a Relational Population Service working set.
  (This does not apply to standard Greentree, which uses *full* RPS.)
- Jade applications cannot use single-file instances of the Jade class JadeBytes.
  (The standard Greentree product does not do this.)
- Classes cannot be configured to use Jade's auto-partition facility.
  (A change in release 2019.4 ensures Greentree's compliance with this.)

There are technical and management pre-requisites:

- Sites must use Microsoft’s Active Directory service. (Jade Help)

Recommendations

Review the costs and benefits of whole database encryption with each site’s stakeholders.

Plan, agree and document your policies and procedures for secure handling of keys and passphrases.

Test the setup using a copy of a production database. Apply these configuration settings:

- Encrypt the whole database rather than individual database (map) files.
- Apply the Default Security access check option.

Be sure to test manual start-up and shut-down, automatic server restart, backup, and recovery from backup.

Warning

Secure handling of keys and passphrases is critical. If you lose them and have encrypted the database using default security, you will be able to use the system on the same server, but you will not be able to run it on another server, and you will not be able to decrypt the database. MYOB and JADE cannot retrieve lost keys or decrypt databases.

Instructions

Follow these Jade instructions:

- Set up server and client nodes to run under Active Directory accounts
- Set the Service Principal Name for the database server in the client INI file
- Encrypting a Database
- Decrypting a Database

Configure Attachment Security

Greentree has a common facility for attaching files to objects in the database. Files can be uploaded using the Windows client, Browser client and via other interfaces. The system itself also creates attachments. Users with suitable permissions can download these attachments and open them on their workstations. There's a risk that these attachments may contain malware which can be activated when users download and open the files.

Greentree can be configured to store attachments *internally* i.e. within the database or *externally* i.e. in a chosen directory on the server. Sites can ensure the safety of attachments by having Greentree store attachments externally, and by configuring regular virus scans of files in the attachments directory.

Recommendations

Configure in Greentree the list of restricted file name extension to suit your site.

Set up Greentree to store attachments externally, in a directory on the server. Configure the directory without execute permission and specify it as an additional directory to be included in backups. In addition, configure anti-virus software to scan files on the directory.

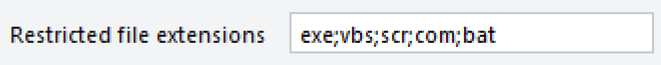
Instructions

The settings for attachments are system-wide but can be viewed and set in any of these places:

- CRM > System > Module Control, Main
- CRM > System > System Options, Email/Attachments
- HR > System > System Options, Email/Attachments
- Workflow > System > Module Control, Other.

To view or change blocked extensions:

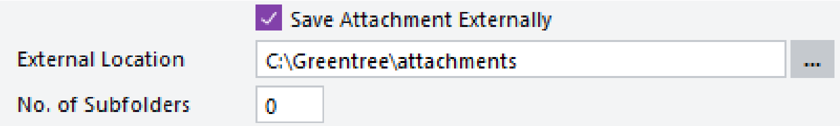
1. Locate the text box for **Restricted file extensions**
2. Add further extensions to the list, separated by semi-colons.
3. Remove extensions if you want to allow some usually restricted file types.



Restricted file extensions: exe;vbs;scr;com;bat

To change from storing attachments internally to storing externally:

1. Locate the check box **Save Attachments Externally** and turn it on.
2. Set the **External Location** to the server directory for attachments.
3. Confirm that you would like to copy all existing attachments to the new location.



☑ Save Attachment Externally

External Location: C:\Greentree\attachments

No. of Subfolders: 0

To set permissions on the attachments directory:

1. Locate the directory in File Explorer.

2. Right-click for Properties and go to the Security tab.
3. Disallow “Read & execute” permission for all users. If necessary, turn off inherited permissions for this directory.
4. Ensure the service account used for Greentree has read and write access.
5. Apply that change to subdirectories.