Mediaprotect
User manual

1 Introduction

Mediaprotect helps you protect media in the same simple way as you protect documents. Once the package is installed and you protect certain media, the files will be password protected. Only authenticated members can open the files, otherwise members will be redirected to the login screen.

Mediaprotect is using the standard Umbraco MemberService, so you can use the existing login views for authentication. And you can plug in your own authentication mechanism.

2 Installation

Install Mediaprotect via CLI or NuGet.

Make sure that the installer has modify rights on the following folders:

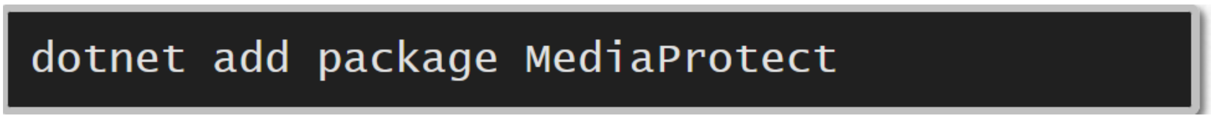
- **/bin**
- **/App_plugins**

The installer also needs rights to create tables in the database.

If for some reason you can’t give the installer these rights, you can install the package manually. Read the chapter manual configuration how to do this.

2.1 CLI

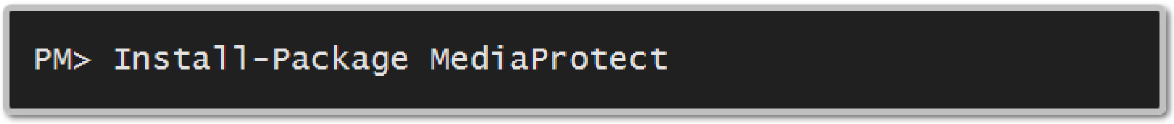
Use the following command to install via CLI.



```
dotnet add package MediaProtect
```

2.2 NuGet

Use the following command in the Package manager console.



```
PM> Install-Package MediaProtect
```

2.3 Add License

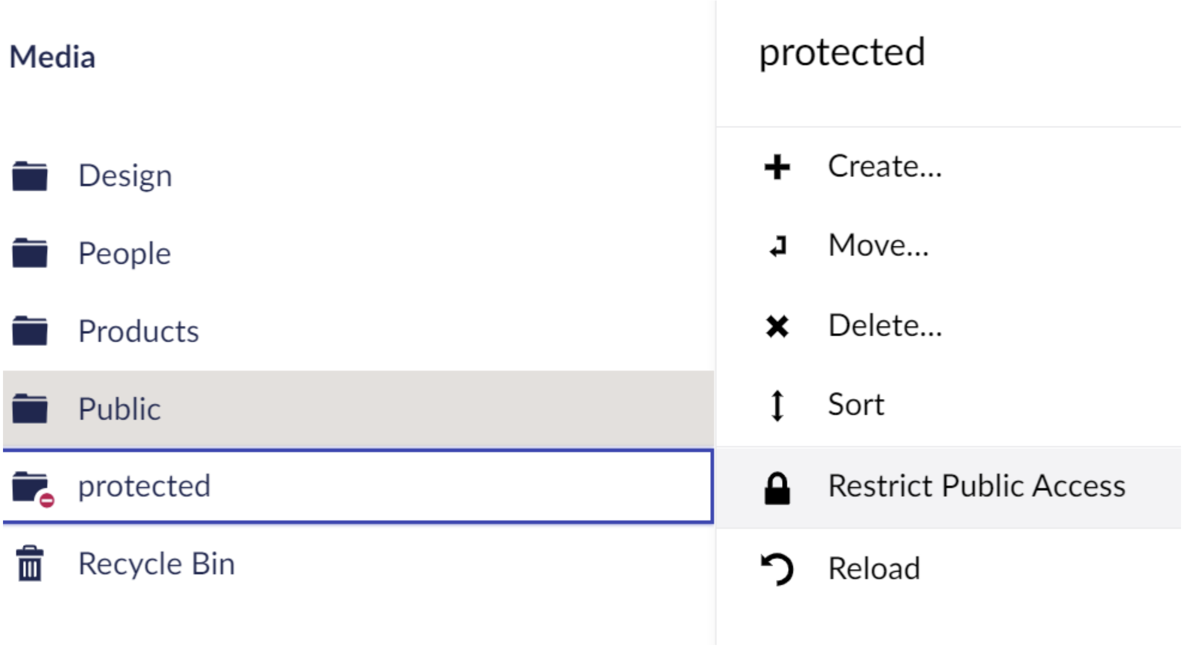
When you purchased a license you can add it to the /Umbraco/Licenses folder and it will automatically be picked up after starting the site.

3 Protect media

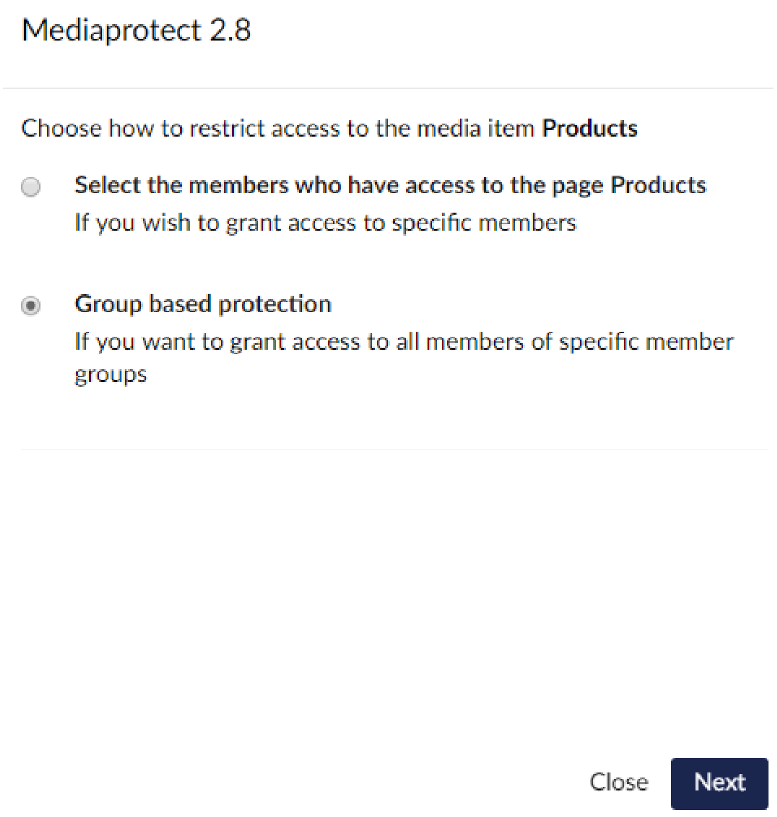
Once the package is installed you can protect your media items.

3.1 Protect media items

You can protect your media items by navigating to the media section, right click on the media folder you want to protect and select Public Access



This will open the same dialog you get when protecting content items.



Mediaprotect 2.8

Choose how to restrict access to the media item **Products**

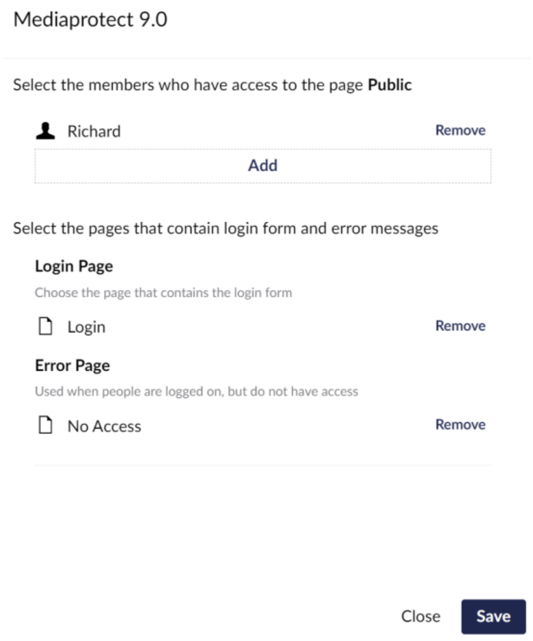
○ **Select the members who have access to the page Products**
If you wish to grant access to specific members

● **Group based protection**
If you want to grant access to all members of specific member groups

Single user protection will protected a media node based on an username and password, role based protection allows you to select one or more roles that have access to the media node.

3.1.1 Single User Protection

When you select single user protection you can specify the username and password. The user will be created with the credentials when you select update. You can also specify the login page where the website visitor will be redirected to when not logged in. And you can specify an error page where the member will be redirected to when the member is not authorized to access the requested file.



Mediaprotect 9.0

Select the members who have access to the page **Public**

Richard Remove

Add

Select the pages that contain login form and error messages

Login Page
Choose the page that contains the login form

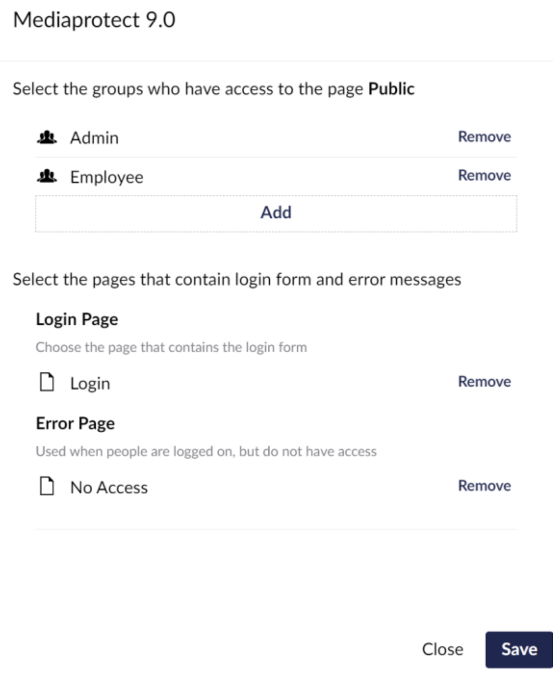
Login Remove

Error Page
Used when people are logged on, but do not have access

No Access Remove

3.1.2 Role based protection

When you select role protection you can specify one or more roles that are allowed to access the media items. You can also specify the login page where the website visitor will be redirected to when not logged in. And you can specify an error page where the member will be redirected to when the member is not authorized to access the requested file.



Mediaprotect 9.0

Select the groups who have access to the page **Public**

Admin Remove

Employee Remove

Add

Select the pages that contain login form and error messages

Login Page

Choose the page that contains the login form

Login Remove

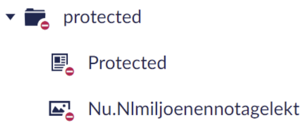
Error Page

Used when people are logged on, but do not have access

No Access Remove

3.1.3 Verify protection

You can recognize a protected media item in the tree or any supported media picker by the no entry sign.



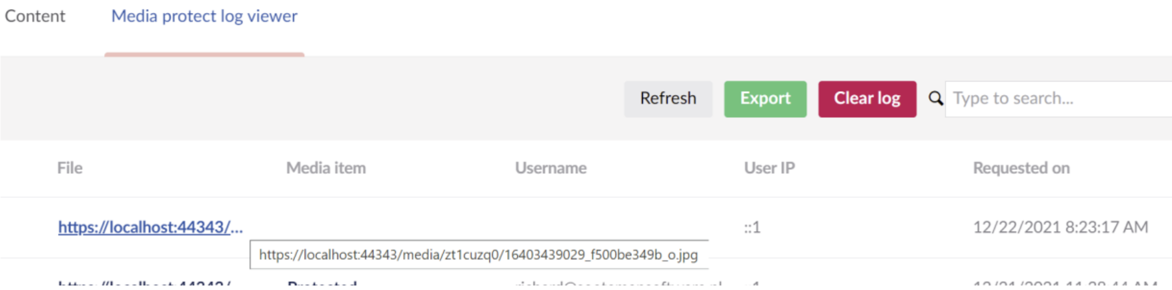
When you open the media item and click on the link you should be redirected to the page you have specified as the login page. If you are already logged in as a website member you should be able to either open the protected media item or being redirected to the not authorized page.

4 Dashboards

4.1 Log protected files requests

During install a dashboard will be installed that enables logging in the media section.

This Dashboard will give you information about which file got requested, the Media item, User info and when the file was requested. You can refresh the overview, export to csv or delete all records.

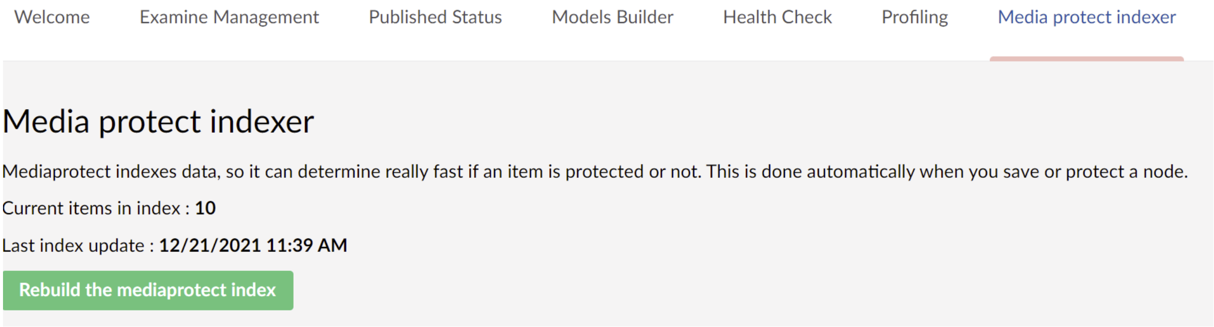


File	Media item	Username	User IP	Requested on
https://localhost:44343/...			::1	12/22/2021 8:23:17 AM
https://localhost:44343/...	Protected	richard@soetemansoftware.nl	::1	12/21/2021 11:38:44 AM

https://localhost:44343/media/zt1cuzq0/16403439029_f500be349b_o.jpg

4.2 Index Dashboard

Media protect uses an index for fast query of media files. This index gets updated when you set protection or save a media Item. In case you ever want to check the last index date, or want to completely rebuild the index you can use the Media Index Dashboard on the developer section.



Media protect indexer

Mediaprotect indexes data, so it can determine really fast if an item is protected or not. This is done automatically when you save or protect a node.

Current items in index : **10**

Last index update : **12/21/2021 11:39 AM**

5 Library

Media protect comes with a small library that you can use in your Views and/or .net libraries to determine if certain items are protected , or who has access to those items.

5.1 Register the helper

If you need to use the library Helper in your view you can inject the helper into your view by adding it to the **_ViewImports.cshtml** file.

```
@inject MediaProtect.Library.MediaProtectHelper MediaProtectHelper
```

Now you can access the Mediaprotect library methods in your view by **MediaProtectHelper.**

5.1.1 MediaIsProtected

Use this extension method to check if a media item is protected based in the Id of the node

```
@MediaProtectHelper.MediaIsProtected(1068)
```

5.1.2 MediaIsProtected

Use this extension method to check if a media item is protected based in thefilename

```
@MediaProtectHelper.MediaIsProtected("/media/eyey/test.pdf")
```

5.1.3 MediaHasAccess

Use this extension method to check if the current logged in user has access to the media item, based on the id of the node.

```
@MediaProtectHelper.MediaHasAccess(1068)
```

5.1.4 MediaHasAccess

Use this extension method to check if the current logged in user has access to the media item, based on the filename .

```
@MediaProtectHelper.MediaHasAccess("/media/eyey/test.pdf")
```

5.1.5 AllowedGroups

Returns a list of groups that have access to the media item.

```
@MediaProtectHelper.AllowedGroups(1068)
```

5.1.6 AllowedGroups

Returns a list of groups that have access to the media file.

```
@MediaProtectHelper.AllowedGroups("/media/eyey/test.pdf")
```

5.1.7 AllowedMembers

Returns a list of usernames that have access to the media item.

```
@MediaProtectHelper.AllowedMembers(1068)
```

5.1.8 AllowedMembers

Returns a list of usernames that have access to the media file.

```
@MediaProtectHelper.AllowedMembers(“/media/eyey/test.pdf”)
```

5.1.9 IsProtectedByUserName

Returns true when the media is node is protected by username protection.

```
@MediaProtectHelper.IsProtectedByUserName(1068)
```

5.1.10 GetProtectedNodesForUser

Returns a list of nodes that are username protected and where the given user has access to.

```
@MediaProtectHelper.GetProtectedNodesForUser(“testuser”)
```

5.1.11 GetProtectedNodesForRole

Returns a list of nodes that are rolebased protected and where the given role has access to.

```
@MediaProtectHelper.GetProtectedNodesForRole(“admin”)
```

5.2 Protection API

It’s also possible to use the API to modify protection info on nodes using the Protection class.

5.2.1 ProtectMedia

This method adds protection to a media node. Specify username protection or not, nodeId, the loginnode and no access node id.

```
MediaProtect.Library.Protection.ProtectMedia(false,1121,1095, 1095)
```

5.2.2 AddMembershipUserToMedia

This method adds a user to a protected node

```
MediaProtect.Library.Protection.AddMembershipUserToMedia(1121,”user”)
```

5.2.3 AddMembershipRoleToMedia

This method adds a role to a protected node

```
MediaProtect.Library.Protection.AddMembershipRoleToMedia(1121,”role”)
```

5.2.4 RemoveMembershipRoleFromMedia

This method removes a role from a protected media item

```
MediaProtect.Library.Protection.RemoveMembershipRoleFromMedia(1121,”role”)
```

5.2.5 RemoveMembershipUserFromMedia

This method removes a user from a protected media item

```
MediaProtect.Library.Protection.RemoveMembershipUserFromMedia(1121,”user”)
```

5.2.6 RemoveProtection

Removes protection from a media item.

```
MediaProtect.Library.Protection.RemoveProtection(1121)
```

5.3 Notifications

MediaProtect comes with a few Notifications. Notifications are wired up exactly the same as Umbraco Notifications are used. Create a class that implements from INotificationHandler<T>, where T is the notification. Notifications can be found in MediaProtect.Library class

5.3.1 Requesting event

Used when a file request is being processed.
INotificationHandler<FileRequestingNotification>

5.3.2 Requested event

Used when a file request is processed.
INotificationHandler<FileRequestedNotification>

5.3.3 Dialog Events

To determine if an item got protected or unprotected from the UI. Media protect comes with two notifications:

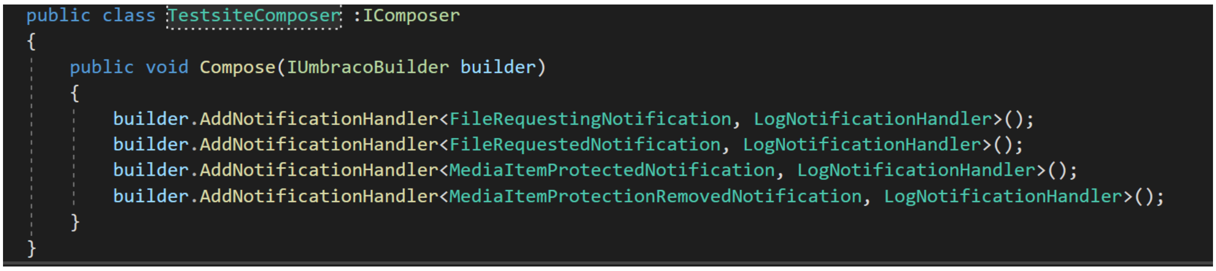
- INotificationHandler<MediaItemProtectedNotification> Will be triggered when a media item gets protected or a protected media item gets updated
- INotificationHandler<MediaItemProtectionRemovedNotification>.Will be triggered when protection is removed from a media item.

5.3.4 Sample

In the sample below you can see all events

5.3.4.1 Composer

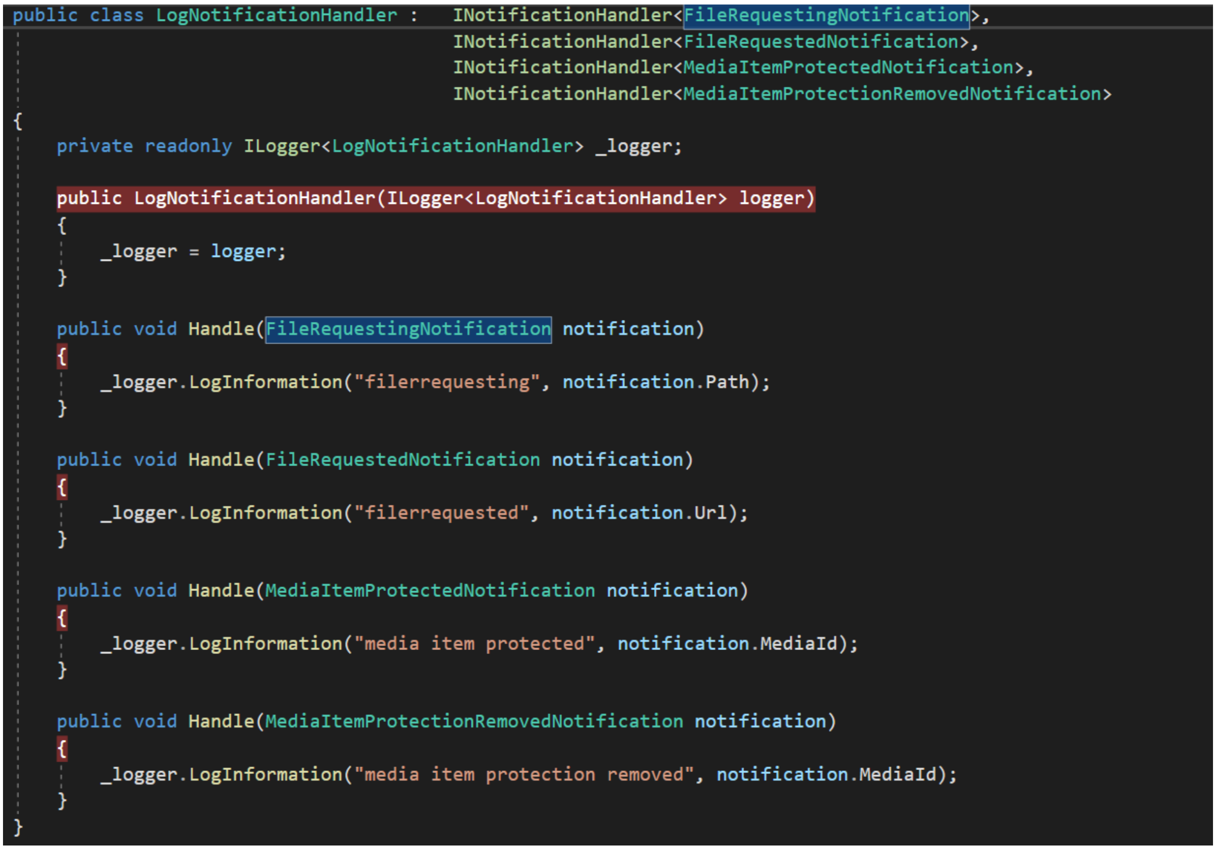
First we need to make sure the notifications are wired up in ur custom composer



```
public class TestsiteComposer :IComposer
{
    public void Compose(IUmbracoBuilder builder)
    {
        builder.AddNotificationHandler<FileRequestingNotification, LogNotificationHandler>();
        builder.AddNotificationHandler<FileRequestedNotification, LogNotificationHandler>();
        builder.AddNotificationHandler<MediaItemProtectedNotification, LogNotificationHandler>();
        builder.AddNotificationHandler<MediaItemProtectionRemovedNotification, LogNotificationHandler>();
    }
}
```

5.3.4.2 The noitficationhandlers

Second we need a class that can log the events



```
public class LogNotificationHandler :   INotificationHandler<FileRequestingNotification>,
                                        INotificationHandler<FileRequestedNotification>,
                                        INotificationHandler<MediaItemProtectedNotification>,
                                        INotificationHandler<MediaItemProtectionRemovedNotification>
{
    private readonly ILogger<LogNotificationHandler> _logger;

    public LogNotificationHandler(ILogger<LogNotificationHandler> logger)
    {
        _logger = logger;
    }

    public void Handle(FileRequestingNotification notification)
    {
        _logger.LogInformation("filerrequesting", notification.Path);
    }

    public void Handle(FileRequestedNotification notification)
    {
        _logger.LogInformation("filerrequested", notification.Url);
    }

    public void Handle(MediaItemProtectedNotification notification)
    {
        _logger.LogInformation("media item protected", notification.MediaId);
    }

    public void Handle(MediaItemProtectionRemovedNotification notification)
    {
        _logger.LogInformation("media item protection removed", notification.MediaId);
    }
}
```

6 Configuration

6.1 Media Protect configuration

When you need to override the default settings you can add the **MediaProtectConfiguration** element to appsettings.json.[1]

```
"MediaProtect": {
   "Enabled": true,
   "AllowAccessForUmbracoUsers": false,
   "EnableLogging": true,
   "LogPublicMedia": true,
   "CsvDelimiter": ",",
   "CsvStringIndicator": "\"",
   "DisableMediaprotectDialog": false,
   "DisableReturnUrl": false,
   "DefaultLoginNode": 1048,
   "DefaultErrorNode": 1049
 }
```

6.1.1 Enable/disable access for logged in Umbraco User

By default a loggedin Umbraco user can access the protected items.

```
"MediaProtectConfiguration": {
   "AllowAccessForUmbracoUsers": false
 }
```

6.1.2 Enable/disable logging

To enable or disable logging set the enableLogging to true/false. It's also possible to log public files, set the logPublicMedia to true to enable logging of public files. Override the default export settings by modifying the csvDelimiter and csvStringIndicator attributes

```
"MediaProtectConfiguration": {
   "EnableLogging": true,
   "LogPublicMedia": true,
   "CsvDelimiter": ",",
   "CsvStringIndicator" :  "\""
 }
```

6.1.3 Disable public access dialog

By default the public access dialog on media is enabled, if you only use the API to modify members/roles you might want to disable the Dialog. Set the value of disableMediaProtectDialog element to true when you want to disable the public access dialog on media

```
"MediaProtectConfiguration": {
   "DisableMediaprotectDialog": false
 }
```

6.1.4 Disable use of returnUrl querystring

By default the returnurl is appended to the url, example ?returnurl=%2fmedia%2f37%2fprotected_file.pdf

[1] Before V13 the section root was MediaProtectConfiguration

If you want to disable this, set the disableReturnUrl config setting to true.

```
"MediaProtectConfiguration": {
   "DisableReturnUrl": true
 }
```

6.1.5 Default login and error node

When you specify default login and error pages these values will be used in the dialog when you protect the node so you don’t have to specify them every time.

```
"MediaProtectConfiguration": {
   "DefaultLoginNode": 1048,
   "DefaultErrorNode": 1049
 }
```

6.1.6 Disable the module without uninstalling

By default Mediaprotect is always on. When you want to disable the module set disabled to true in the config file

```
"MediaProtectConfiguration": {
   "Enabled": true
 }
```

7 Troubleshooting

7.1 I get an Invalid License exception.

Make sure you've bought the correct license for the (sub)domain , or an enterprise license. Contact support@soetemansoftware.nl for help.

7.2 When I browse to the file it's still visible on the website

Make sure Mediaprotect module is running by browsing to /app_plugins/mediaprotect/mediaprotect.txt. This should return true

7.3 How can I create a login box on my login page.

Umbraco comes with some partial templates for login and loginstatus. Use those.

7.4 I have another question

If you have any question please email us at support@soetemansoftware.nl we love to help you.