Extensions

5.X

Extensions are plug-ins to eZ Publish, providing additional custom functionality. Various extensions are available for eZ Publish. All of them require the same basic steps for an installation. This chapter will show how to perform the following:

1. Extract the compressed archive containing the extension
2. Activate the extension

Some extensions might require further action to make them fully functional, e.g. creating new database tables, adding certain content classes to eZ Publish, etc. Such additional measures are explained in the documentation for each extension.

As outlined before, this section deals with the basic steps only. For demonstration purposes, the installation will be exemplified by an imaginary extension called "ezfoo".

Executable packages
This issue affects installations using eZ Publish Legacy, either stand-alone, or as part of eZ Platform 5.x, or in eZ Platform 1.11 and newer using LegacyBridge. If you are not using Legacy in any way, you are not affected.

The package system, by design, allows you to package an extension into a file, and export/import such packages. Extensions can of course contain PHP scripts, and they usually do. Such scripts can be used in an attack on the server. This problem is fundamental and cannot be fixed by any other means than by removing the feature.

By default, only the Administrator has the permissions to use the package system. It follows that the Administrator role, and any others granted packaging permissions, can only be held by users who already have access to the server, and/or can be trusted not to exploit this access.

As a consequence eZ Publish legacy should not be used in the type of shared hosting installation where Administrators are not supposed to have access to the underlying operating system, or to other eZ Publish installations on the same server. The package system is an old part of eZ Publish legacy, and it was not designed for that kind of installation. Currently this is not considered best practice anyway - setups using e.g. Docker and Platform.sh allow you to completely separate installations from each other. This is a better way to keep things secure than relying on PHP scripts being read-only even for administrators. (The package system does not exist in eZ Platform and will not be added there, since extensions are not used there.)

In summary:

If you are responsible for legacy installations where administrators cannot be fully trusted not to exploit their privileges, make sure to properly lock down the package system and/or fully separate web sites from each other.
Make sure that the administrator password(s) are secure, and not using the default administrator password.

Proposed quick solution for those affected:

If you are administrating a shared hosting solution of this kind, it may take a while to change the setup. Meanwhile, one quick way to lock down the package system is to use rewrite rules to block all access to package URLs. Apache example:
`RewriteRule ^/package/.* - [R=403,L]`
or with URL-based SiteAccess:
`RewriteRule ^/my_site_access/package/.* - [R=403,L]`
or supporting both cases, and multiple SiteAccesses:
`RewriteRule ^(/my_site_access|/my_site_access_admin)?/package/.* - [R=403,L]`
This can be placed before other rules.

To be absolutely certain you can also (or instead of this) delete the `/kernel/package` directory in the eZ Publish web root. Please note that this will break the legacy installation wizard, since it relies on the package system to install the demo design.

Once the situation is resolved these measures should be reversed, to bring back the package features. You may want to do a review of whether the issue may have been exploited on your server(s).