Security Releases

When potential security holes are discovered in SilverStripe's supported modules (/software/addons/silverstripe-commercially-supported-module-list/), we produce security releases to ensure that you are able to promptly secure your SilverStripe websites (check our security release process (http://docs.silverstripe.org/en/contributing/release_process/)). All releases are available on our download (stable-download/) page, and are announced (https://forum.silverstripe.org/c/releases) on our forums (register to subscribe). Vulnerabilities in releases are disclosed here. Please subscribe to our security release RSS feed (software/download/security-releases/rss) and pre-announcement mailing list (https://docs.silverstripe.org/en/4/contributing/release_process/#pre-announce-mailinglist) to stay updated.

CVE-2022-38724 XSS in shortcodes

Severity:
Medium (? (https://docs.silverstripe.org/en/contributing/release_process/#security-releases))
Identifier:
CVE-2022-38724
Versions Affected:
silverstripe/framework: ^4.0.0, silverstripe/assets: ^1.0.0
Versions Fixed:
silverstripe/framework: ^4.11.13, silverstripe/assets: ^1.11.1
Release Date:
2022-11-21

A malicious content author could add arbitrary attributes to HTML editor shortcodes which could be used to inject a JavaScript payload on the front end of the site. The shortcode providers that ship with Silverstripe CMS have been reviewed and attribute whitelists have been implemented where appropriate to negate this risk.

Most projects should be able to apply the patch without further work. There's no legitimate use case for this behaviour. If your project includes custom shortcode providers (https://docs.silverstripe.org/en/4/developer_guides/extending/shortcodes/#defining-custom-shortcodes), consider reviewing them and implementing a similar whitelist when rendering the shortcodes to HTML.

Regression testing should focus on HTML Editor functionally relying on shortcodes:

- image insertion
- links to CMS resources
- media insertion
- custom shortcodes for your project.

Base CVSS: 4.6 (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:L/UI:R/S:U/C:L/I:L/A:N/E:P/RL:O/RC:C&version=3.1)

Reported by: Steve Boyd (https://www.silverstripe.com/about-us/team/?member=0-59) from Silverstripe Ltd

CVE-2022-38462 Reflected XSS in querystring parameters

Severity:
Medium (? (https://docs.silverstripe.org/en/contributing/release_process/#security-releases))
Identifier:
CVE-2022-38462
Versions Affected:
silverstripe/framework: ^3.0.0, ^4.0.0
Versions Fixed:
silverstripe/framework: 4.11.13
Release Date:
2022-11-21

An attacker could inject a XSS payload in a Silverstripe CMS response by carefully crafting a return URL on a `/dev/build` or `/Security/login` request.

To exploit this vulnerability, an attacker would need to convince a user to follow a link with a malicious payload.

This will only affect projects configured to output PHP warnings to the browser. By default, Silverstripe CMS will only output PHP warnings if your `SS_ENVIRONMENT_TYPE` environment variable is set to dev. Production sites should always set `SS_ENVIRONMENT_TYPE` to `live`.

Read the Environment management (https://docs.silverstripe.org/en/4/getting_started/environment_management/) documentation for more details on configuring environment variables.

Most projects should be able to apply the patch without further work. There's no legitimate use case for this behaviour.

Regression testing should focus on areas where the `location` header is used to redirect users.

Base CVSS: 4.2 (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:L/A:N&version=3.1) (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:L/UI:R/S:U/C:L/I:L/A:N/E:P/RL:O/RC:C&version=3.1)

Reported by: TF1T (https://huntr.dev/users/trungtin1998/) via huntr.dev

CVE-2022-38148 Blind SQL Injection via GridFieldSortableHeader

Severity:
High (? (https://docs.silverstripe.org/en/contributing/release_process/#security-releases))
Identifier:

CVE-2022-38148
Versions Affected:
silverstripe/framework: ^3.0.0, ^4.0.0
Versions Fixed:
silverstripe/framework: 4.10.11, 4.11.14
Release Date:
2022-11-21

Gridfield state is vulnerable to SQL injections. The vast majority of Gridfields in Silverstripe CMS are affected by this vulnerability.

An attacker with CMS access could execute an arbitrary SQL statement by adding an SQL payload in some parts of the GridField state.

Most projects should be able to apply the patch without further work. There's no legitimate use case for this behaviour.

Regression testing should focus on custom filtering and sorting logic for Gridfields.

Base CVSS: 7.1 (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:L/A:N/E:P/RL:O/RC:C&version=3.1) (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:L/UI:R/S:U/C:L/I:L/A:N/E:P/RL:O/RC:C&version=3.1)

Reported by: TF1T (https://huntr.dev/users/trungtin1998/) via huntr.dev

CVE-2022-38147 XSS via uploaded gpx file

Severity:
Medium (? (https://docs.silverstripe.org/en/contributing/release_process/#security-releases))
Identifier:
CVE-2022-38147
Versions Affected:
silverstripe/assets: ^1.0.0
Versions Fixed:
silverstripe/assets: 1.11.1
Release Date:
2022-11-21

A malicious content author could upload a GPX file with a Javascript payload. The payload could then be executed by luring a legitimate user to view the file in a browser with support for GPX files. GPX is an XML-based format (https://fileinfo.com/extension/gpx) used to store GPS data.

By default, Silverstripe CMS will no longer allow GPX files to be uploaded to the assets area.

Most projects should be able to apply the patch without further work. While there can be a legitimate use case for using GPX files, it's an uncommon one. You can re-enable support for GPX files (https://docs.silverstripe.org/en/4/developer_guides/files/allowed_file_types/#file-extensions-validation) if you have a need for them, but beware there's an inherent risk in allowing content authors to upload this kind of file.

Regression testing should focus on identifying if your site makes use of any GPX files. You can validate if you have any pre-existing GPX file on your Silverstripe CMS site by accessing the Files area and searching for "GPX". You'll want to delete any GPX file prior to applying the patch.

Base CVSS: 4.6 (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:L/UI:R/S:U/C:L/I:L/A:N/E:P/RL:O/RC:C&version=3.1) (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:H/PR:L/UI:R/S:U/C:L/I:L/A:N&version=3.1)

Reported by: nhienit (https://twitter.com/__nhienit__) via huntr.dev

CVE-2022-38146 URL XSS vulnerability due to outdated jquery in CMS

Severity:
Medium (? (https://docs.silverstripe.org/en/contributing/release_process/#security-releases))
Identifier:
CVE-2022-38146
Versions Affected:
silverstripe/admin: ^1.0.0
Versions Fixed:
silverstripe/admin: ^1.11.3
Release Date:
2022-11-21

The Silverstripe CMS UI uses jQuery 1.7.2. This version of jQuery is affected by CVE-2019-11358 Object.prototype pollution (https://nvd.nist.gov/vuln/detail/cve-2019-11358). An attacker could perform an XSS attack by convincing a user to follow a link with a specially crafted `__proto__` query string parameter.

silverstripe/admin 1.11.3 addresses this problem by stopping all JavaScript execution if a `__proto__` query string is present in the URL. This fix is just a stopped gap measure.

This issue will be properly remediated by upgrading to jQuery 3.6.1 or later in the Silverstripe CMS 4.12.0 release.

Most projects should be able to apply the patch without further work. There's no legitimate use case for this behaviour.

Regression testing should focus on custom CMS UI functionality that might be implemented in your project.

If you use the jQuery version distributed with Silverstripe CMS in the front end of your site, you may be affected by this vulnerability via the front end. If this applies to youp project, you should upgrade your theme to use a more recent jQuery version.

Base CVSS: 5.4 (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:N/UI:R/S:U/C:L/I:L/A:N/E:P/RL:O/RC:C&version=3.1)

Reported by: Trong Pham (https://huntr.dev/users/dtroo/) via huntr.dev

CVE-2022-38145 Stored XSS in Compare Mode

Severity:
Medium (? (https://docs.silverstripe.org/en/contributing/release_process/#security-releases))
Identifier:
CVE-2022-38145
Versions Affected:
silverstripe/versioned-admin: ^1.0.0
Versions Fixed:
silverstripe/versioned-admin: ^1.11.1
Release Date:
2022-11-21

A malicious content author could add a Javascript payload to a page's meta description and get it executed in the versioned history compare view.

This vulnerability requires access to the CMS to be deployed. The attacker must then convince a privileged user to access the version history for that page.

Most projects should be able to apply the patch without further work. There's no legitimate use case for this behaviour.

Regression testing should focus on version comparison with the page history tab.

Base CVSS: 4.6 (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:L/UI:R/S:U/C:L/I:L/A:N/E:P/RL:O/RC:C&version=3.1)

Reported by: TF1T (https://huntr.dev/users/trungtin1998/) via huntr.dev

CVE-2022-37430 Stored XSS using uppercase characters in HTMLEditor

Severity:
Medium (? (https://docs.silverstripe.org/en/contributing/release_process/#security-releases))
Identifier:
CVE-2022-37430
Versions Affected:
silverstripe/framework: ^3.0.0, ^4.0.0
Versions Fixed:
silverstripe/framework: 4.11.13
Release Date:
2022-11-21

A malicious content author could add a Javascript payload to the `href` attribute of a link. A similar issue was identified and fixed via CVE-2022-28803 (/download/security-releases/cve-2022-28803/). However, the fix didn't account for the casing of the `href` attribute.
An attacker must have access to the CMS to exploit this issue.

Most projects should be able to apply the patch without further work. There's no legitimate use case for this behaviour.

Regression testing should focus on link creations within HTML editor fields.

Base CVSS: 4.6 (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:L/UI:R/S:U/C:L/I:L/A:N/E:P/RL:O/RC:C&version=3.1)

Reported by: Steve Boyd (https://www.silverstripe.com/about-us/team/?member=0-59) from Silverstripe Ltd

CVE-2022-37429 Stored XSS using HTMLEditor

Severity:
Medium (? (https://docs.silverstripe.org/en/contributing/release_process/#security-releases))
Identifier:
CVE-2022-37429
Versions Affected:
silverstripe/framework: ^4.0.0, ^3.0.0
Versions Fixed:
silverstripe/framework: 4.11.13
Release Date:
2022-11-21

A malicious content author could add a JavaScript payload to the `href` attribute of a link by splitting a `javascript` URL with white space characters.

An attacker must have access to the CMS to exploit this issue.

Most projects should be able to apply the patch without further work. There's no legitimate use case for this behaviour.

Regression testing should focus on link creations within HTML editor fields.

Base CVSS: 4.6 (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:L/UI:R/S:U/C:L/I:L/A:N/E:P/RL:O/RC:C&version=3.1)

Reported by: TF1T (https://huntr.dev/users/trungtin1998/) via huntr.dev

CVE-2022-37421 Stored XSS in custom meta tags

Severity:
Low (? (https://docs.silverstripe.org/en/contributing/release_process/#security-releases))
Identifier:
CVE-2022-37421
Versions Affected:
silverstripe/cms: ^4.0.0, ^3.0.0
Versions Fixed:
silverstripe/cms: 4.11.3
Release Date:

2022-11-21

A malicious content author could create a custom meta tag and execute an arbitrary JavaScript payload. This would require convincing a legitimate user to access a page and enter a custom keyboard shortcut. This requires CMS access to exploit.

Most projects should be able to apply the patch without further work. There's no legitimate use case for this behaviour.

Regression testing should focus on pages with pre-existing custom meta tags, if any are present.

Base CVSS: 3.7 (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:H/PR:L/UI:R/S:U/C:L/I:L/A:N&version=3.1)

Reported by: TF1T (https://huntr.dev/users/trungtin1998/) via huntr.dev

CVE-2022-29858 - Unpublished, protected files can be published via shortcode

Severity:
Medium (? (https://docs.silverstripe.org/en/contributing/release_process/#security-releases))
Identifier:
CVE-2022-29858
Versions Affected:
silverstripe/assets: <=1.10.0
Versions Fixed:
silverstripe/assets: 1.10.1
Release Date:
2022-06-28

Draft protected images can be published by changing an existing image shortcode on website content to match the ID of the draft protected image and then publishing the website content.

Base CVSS: 4.3 (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:N/A:N/E:P/RL:O/RC:C&version=3.1)

Reported by: ranjit-git via huntr.dev

CVE-2022-28803 - Stored XSS in link tags added via XHR

Severity:
Medium (? (https://docs.silverstripe.org/en/contributing/release_process/#security-releases))
Identifier:
CVE-2022-28803
Versions Affected:
silverstripe/framework: <=4.10.8
Versions Fixed:
silverstripe/framework: 4.10.9
Release Date:
2022-06-28

XSS inside the href attribute of an HTML hyperlink can be added to website content via XHR by an authenticated CMS user.

Base CVSS: 5.4 (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:L/A:N/E:P/RL:O/RC:C&version=3.1)

Reported by: ranjit-git via huntr.dev

CVE-2022-25238 - Stored XSS via HTML fields

Severity:
Medium (? (https://docs.silverstripe.org/en/contributing/release_process/#security-releases))
Identifier:
CVE-2022-25238
Versions Affected:
silverstripe/framework: <=4.10.8
Versions Fixed:
silverstripe/framework: 4.10.9
Release Date:
2022-06-28

XSS inside of script tags can can be added to website content via XHR by an authenticated CMS user if the cwp-core module is not installed on the sanitise_server_side contig is not set to true in project code.

Base CVSS: 5.4 (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:L/A:N/E:P/RL:O/RC:C&version=3.1)

Reported by: Greg Best from Aura Information Security (https://www.aurainfosec.com/)

CVE-2022-24444 - Hybridsessions does not expire session id on logout

Severity:
Medium (? (https://docs.silverstripe.org/en/contributing/release_process/#security-releases))
Identifier:
CVE-2022-24444
Versions Affected:
silverstripe/hybridsessions: <=2.4.0, 2.5.0

Versions Fixed:
silverstripe/hybridsessions: 2.4.1, 2.5.1
Release Date:
2022-06-28

When using the hybridsessions module is used without the session-manager module installed and sessions IDs are saved to disk, unexpired SessionIDs of logged out users can still be used to make authenticated requests.

Base CVSS: 4.8 (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:H/PR:N/UI:N/S:U/C:L/I:L/A:N/E:P/RL:O/RC:C&version=3.1)

Reported by: Kartik Patel

CVE-2021-41559 - Quadratic blowup in Convert::xml2array()

Severity:
Medium (? (https://docs.silverstripe.org/en/contributing/release_process/#security-releases))
Identifier:
CVE-2021-41559
Versions Affected:
silverstripe/framework: <=4.10.8
Versions Fixed:
silverstripe/framework: 4.10.9
Release Date:
2022-06-28

The Convert::xml2array() function is vulnerable to quadratic blowup where a malicious xml doctype with internal entities can cause CPU usage to go to 100% and stay there.

Base CVSS: 4.8 (https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator?vector=AV:N/AC:H/PR:L/UI:R/S:U/C:N/I:N/A:H/E:U/RL:O/RC:R&version=3.1)

Reported by: Matthew Dekker from ZX Security

CVE-2022-29254 Failed payment recorded has completed

Severity:
Low (? (https://docs.silverstripe.org/en/contributing/release_process/#security-releases))
Identifier:
CVE-2022-29254
Versions Affected:
silverstripe/silverstripe-omnipay: <=2.5.1, <=3.0.1, <=3.1.3, <=3.2.0
Versions Fixed:
silverstripe/silverstripe-omnipay: 2.5.2, 3.0.2, 3.1.4, 3.2.1
Release Date:
2022-06-06

For a subset of Omnipay gateways (those that use intermediary states like `isNotification()` or `isRedirect()`), if the payment identifier or success URL is exposed it is possible for payments to be prematurely marked as completed without payment being taken. This is mitigated by the fact that most payment gateways hide this information from users, however some issuing banks offer flawed 3DSecure implementations that may inadvertently expose this data.