

9 Changelog

9.1 Version 12 / February 2021

Version 12 introduces encoding for PHP8 - a great new version of PHP for years. We were glad to receive comments and suggestions and want to thank you very much for sharing your ideas! We are looking forward to hearing about other suggestions for improving SourceGuardian. Here is a list of our version 12 changes.

- Full support of PHP8 encoding including all the latest language features. Note, files encoded with older versions of SourceGuardian need to be re-encoded with SourceGuardian 12 in order to run under PHP8.
- New fully supported PHP8 language features include: named parameters, attributes, union types, mixed pseudo type, null-safe operator, match expressions, using throw in expressions, constructor properties and others.
- Due to native support for attributes (annotations) in PHP8, PHP frameworks which uses annotations like Symfony must work with encoded files out of the box (after the frameworks get updated accordingly by their authors).
- New loaders for PHP8. Please check [our blog](#) and the [loaders page](#) for new versions

We are happy to work with our customers if they need bespoke loaders for other custom operating system. Please contact support@sourceguardian.com if you are interested.

We do not include loaders for all supported operating systems to the GUI package. Please visit our [loaders page](#) if you can't find the loader for your OS in the [Copy Loaders](#) dialog.

New Features

- Multiple external license file names may be specified when locking to a license. Separate license file names, paths or URLs by a comma in the `--external` or an appropriate GUI option. Please note, the license files will be checked in the order they are specified. You may mix local file names, paths or URLs. If neither of the license files is found and running of the protected file fails with a "a license file is required" error, the error message will include only the first license file name. This is normal and by design.
- `sg_get_const()` if called without parameters returns an array of all the constants.
- locking to a machine id is refined for Linux. Please note, Linux lacks of a strict machine id algorithm. If you are getting unstable machine id result when running protected scripts on Linux, consider locking to MAC address(es) instead of machine id as an alternative approach.

Update for PHP 7.x code

- We have fixed some critical issues in PHP 7.x family of encoders and loaders. This means files encoded with version 12 of SourceGuardian for PHP 7.x will not run with older loaders. If you use version 12 of SourceGuardian, please make sure you install and/or deploy the recent loaders.

9.2 Version 11.4 / February 2020

Version 11.4 introduces encoding for PHP 7.4. We were glad to receive comments and suggestions and want to thank you very much for sharing your ideas! We are looking forward to hearing about other suggestions for improving SourceGuardian. Here is a list of all the version 11.4 changes.

- Full support of PHP 7.4 encoding including all the latest language options. Note, files encoded with older versions of SourceGuardian need to be re-encoded with SourceGuardian 11.4 in order to run under PHP 7.4
- New loaders for PHP 7.4. Please check [our blog](#) and the [loaders page](#) for new versions

We are happy to work with our customers if they need bespoke loaders for other custom operating system. Please contact support@sourceguardian.com if you are interested.

We do not include loaders for all supported operating systems to the GUI package. Please visit our [loaders page](#) if you can't find the loader for your OS in the [Copy Loaders](#) dialog.

New Features

- New `--ignore-symlinks` option for the CLI version of the encoder to skip symlinks while recursively scanning the folders.

GUI updates

- Some visual changes for Lock and Preferences screens.

9.3 Version 11.3 / April 2019

Version 11.3 introduces encoding for PHP 7.3 and many new features. We were glad to receive comments and suggestions and want to thank you very much for sharing your ideas! We are looking forward to hearing about other suggestions for improving SourceGuardian. Here is a list of all the version 11.3 changes.

- Full support of PHP 7.3 encoding including all the latest language options.
- We fixed some issues with unexpected segfaults on PHP7+ and some other problems. If you experienced issues like that with your code, please re-encode with the latest version 11.3 of

SourceGuardian and install the recent loader.

Note, files encoded with SourceGuardian 11.0, 10.x or older need to be re-encoded with SourceGuardian 11.3 in order to run them under PHP 7.3

- New loaders for PHP 7.3, updated loaders for older versions of PHP

We updated loaders and created new ones for PHP 7.3. Loaders for the following operating systems are available:

Windows 32-bit (VC6, VC9; VC11 PHP 5.5, 5.6; VC14 PHP 7.0, 7.1; VC15 PHP 7.2)
Windows 64-bit (VC11 PHP 5.5, 5.6; VC14 PHP 7.0, 7.1, VC15 PHP 7.2)
MacOSX (universal binaries, include i386, x86_64)
Linux (i386, x86_64)

We update the following loaders on request. Please check [our blog](#) and the [loaders page](#) for new versions.

FreeBSD (i386, x86_64)
Linux ARM (armel)
Linux ARM (armhf) Raspberry Pi (including Pi version 3) and other boards

We are happy to work with our customers if they need bespoke loaders for other custom operating system. Please contact support@sourceguardian.com if you are interested.

We do not include loaders for all supported operating systems to the GUI package. Please visit our [loaders page](#) if you can't find the loader for your OS in the [Copy Loaders](#) dialog.

New Features

- Experimental licensing for Docker. If you are installing SourceGuardian to your Docker, licensing must work correctly now letting you install and use SourceGuardian on this Docker machine. So, if you are using SourceGuardian from a Docker container during the deployment process of your files, this must work now since you installed and registered your copy of SourceGuardian as usual. Note, every separate installation of SourceGuardian still requires an additional license as one license lets you install and use SourceGuardian only on one machine.

Installing to Docker needs a special approach and mapping of /var/run/docker.sock Please refer to a new section [Installing to Docker](#) in the user manual.

- We are introducing a new locking option which is available in the full version of the encoder - locking to a machine ID. Use the new loader method `sg_get_machine_id()` to obtain a machine ID on your client machine and then specify the machine ID on the Lock screen in GUI or use the new command line option `--machine-id` of the encoder or licgen tool. Encoded PHP scripts locked to a machine ID will only run on that machine (or machines if you specify multiple machine IDs). Please refer to the [Locking options](#) section for further information.

Note: as `sg_get_machine_id()` is a binary method of the loader, an appropriate loader must be installed to the client machine in order for this method to be available from your code. We recommend that you create a mini project, encode it and run for obtaining the machine ID from the client.

- We are also introducing a new locking option for CLI PHP scripts. It was always a problem to lock

such the scripts as neither IP nor domain locking may be used for them. Locking to MAC address was only a solution in that case, but it's not always convenient to lock to MAC addresses. Now you may use a special verification URL to validate the CLI script and make sure it works on the same machine as your web based part of the project.

For this to work you need to do two things:

1) Create a special PHP script which is accessible via HTTP protocol, it will be used to validate the machine. The only directive you need to put into it is:

```
<?php echo sg_get_verification_id(); ?>
```

Note: if you use any frameworks, you may need to use another mechanism for sending a string to the output instead of *echo*

2) When encoding your CLI script use the new `--remote-verification-url` option to specify the full URL to your web verification script and use the same project ID as for the web part of your code and particularly the verification script.

Note 1: as the `--remote-verification-url` is mostly used only for locking of the encoded CLI scripts, consider separate encoding of the web part of your project and CLI scripts. You may create two GUI projects for that or use the command line encoder. Use the same project IDs/Keys for both!

Note 2: You may use standard locking to IP or domain for your web verification script (as usual for this to work your web server must send the information about current IP and domain to PHP via environment). Anyway, you "lock" to the domain if use the domain name in the verification URL or lock to IP if you use IP in the URL - choose the way which suits your project and the deployment process.

However, if your CLI PHP script works on its own and is not a part your your web based project, then you still may use the new [machine ID locking](#) option for it as well as good old locking to MAC addresses.

Please refer to the [Locking options](#) section for further information about using of the remote verification URL option.

- Option to add custom auto-globals. For CLI use: `--auto-global MYAUTO` If you use GUI, add this CLI option to 'additional command line options' in [Advanced options](#).
- Options to ignore IP/domain check for the scripts running with CLI PHP. It means if you have encoded the entire project with locking to a IP/domain and selected the option to ignore the IP/domain check for CLI, your protected files will require to be run under the specified IP/domain(s) for web but the same files will be run OK with CLI PHP. This simplifies running PHP CLI scripts like cron jobs but still have them encoded in the same project along with web scripts. However, you may consider the risk of using this option as it does what it does - lets run your IP/domain locked encoded files with PHP CLI bypassing the domain check. For CLI use: `--ip-ignore-cli`, `--domain-ignore-cli` or the appropriate tickboxes on the [Lock screen](#) in GUI. Regardless these are the new options, please consider locking CLI scripts to the [verification URL](#) instead.
- PHP doc comments are now removed by default. Use `--keep-doc-comments` option to keep them e.g. if the framework you use require doc comments. We added an appropriate tickbox to the [Advanced options](#) window in GUI.
- Error text for `sourceguardian.restrict_unencoded=1` updated to include the file name

Error code [08] text changed to:

PHP Fatal error: SourceGuardian Loader - This script ... or other unencoded scripts cannot run in this environment. Please contact the author of the script regarding this problem. [08]

Error code [15] text changed to:

PHP Fatal error: SourceGuardian Loader - This protected script ... can run only in conjunction with other encoded files of the same project. Please contact the author of the script regarding this problem. Error code [15]

GUI updates

- File associations work now on Windows and macOS. It means clicking a .sg project file in Explorer/ Finder will launch SourceGuardian and opens this project in GUI. Note, for this to work, SourceGuardian must be installed using the installer on Windows, on macOS, you need to launch SourceGuardian at least once for file associations to start working.
- Hex registration code is now displayed in Help/Registration information which is useful if you are using multiple installations of SourceGuardian and need to manage or reset a particular license in the online user profile, now you may easily find it there.
- You may access your SourceGuardian online user profile directly from the application, click Help/ SourceGuardian User Profile.

CLI updates

- --days option was replaced with --expire 00d 00m 00h 00s
- New command line options described above: --ip-ignore-cli, --domain-ignore-cli, --machine-id, --machine-id-encrypt, --remote-verification-url, --keep-doc-comments, --auto-global

Bug Fixes

- using self:: from within nested function which in its turn is defined in a private member was not working

9.4 Version 11.2 / March 2018

Version 11.2 introduces encoding for PHP 7.2 and other new options for the command line encoder. We were glad to receive comments and suggestions and want to thank you very much for sharing your ideas! We are looking forward to hearing about other suggestions for improving SourceGuardian. Here is a list of all the version 11.2 changes.

- Full support of PHP 7.2 encoding including all the latest language options: converting numeric keys in object/array casts, counting of non-countable objects, object typehint, new optimized opcodes.
- We fixed some issues with unexpected segfaults on 7.1 and some other problems. If you experienced

issues like that with your code, please re-encode with the latest version 11.2 of SourceGuardian and install the recent loader.

Note, files encoded with SourceGuardian 11.0, 10.x or older need to be re-encoded with SourceGuardian 11.2 in order to run them under PHP 7.2

- New loaders for PHP 7.2, updated loaders for older versions of PHP

We updated loaders and created new ones for PHP 7.2. Loaders for the following operating systems are available:

Windows 32-bit (VC6, VC9; VC11 PHP 5.5, 5.6; VC14 PHP 7.0, 7.1; VC15 PHP 7.2)
 Windows 64-bit (VC11 PHP 5.5, 5.6; VC14 PHP 7.0, 7.1, VC15 PHP 7.2)
 MacOSX (universal binaries, include i386, x86_64)
 Linux (i386, x86_64)

We update the following loaders on request. Please check [our blog](#) and the [loaders page](#) for new versions.

FreeBSD (i386, x86_64)
 Linux ARM (armel)
 Linux ARM (armhf) Raspberry Pi (including Pi version 3) and other boards

We are happy to work with our customers if they need bespoke loaders for other custom operating system. Please contact support@sourceguardian.com if you are interested.

We do not include loaders for all supported operating systems to the GUI package. Please visit our [loaders page](#) if you can't find the loader for your OS in the [Copy Loaders](#) dialog.

Command line encoder updates

- A minor addition which may be a great help for the users who encode from the command line. We added "+" and "-" options for --phpversion. "+" means to encode for the specified version of PHP and for all the newer versions which are supported by the current version of SourceGuardian. "-" means to encode for all the supported versions of PHP except the specified one and all the lower versions, which is useful if you always need to encode by default for new versions but do not need support for old versions starting from some one. E.g.

--phpversion 5.6+ version of SourceGuardian)	encodes for PHP 5.6 and all the newer versions (up to 7.2 for this version of SourceGuardian)
--phpversion 7.0+ SourceGuardian)	encodes for PHP 7.0, 7.1, 7.2 (up to 7.2 for this version of SourceGuardian)
--phpversion 5.4- --phpversion 4-	encodes for PHP 5.5 and newer, i.e. excludes PHP 5.4 and older excludes PHP 4 completely, encodes for all PHP 5+, PHP 7+

As usual when encoding for multiple versions of PHP please make sure your code is compatible with ALL the selected versions of PHP, otherwise the encoder displays an error and that source PHP file may remain unencoded in the target folder.

9.5 Version 11.1 / April 2017

Version 11.1 introduces encoding for PHP 7.1, updated GUI and other new options. Although it looks like a minor update, in fact it's bigger than usual. We did serious work on refining GUI and adding some new and useful options to the command line version. As usual this update is partly based on comments and suggestions of our users. We were glad to receive comments and suggestions and want to thank you very much for sharing your ideas! We are looking forward to hearing about other suggestions for improving SourceGuardian. Here is a list of all the version 11.1 changes.

- Full support of PHP 7.1 encoding including all the latest language options: nullable types, void functions, class constants visibility, iterable, multi-catch and more.

Encoding for PHP 7.1 is fully supporting in version 11.1 of SourceGuardian. PHP 7.1 introduced new language features and updated bytecode format to support them. Files encoded with SourceGuardian 11.0, 10.x or older need to be re-encoded with SourceGuardian 11.1 in order to run protected files under PHP 7.1

- New loaders for PHP 7.1, updated loaders for older versions of PHP

We updated loaders and created new ones for PHP 7.1. Loaders for the following operating systems are available:

Windows 32-bit (VC6, VC9; VC11 PHP 5.5, 5.6; VC14 PHP 7.x)
Windows 64-bit (VC9 PHP 5.3,5.4; VC11 PHP 5.5,5.6; VC14 PHP7.x)
MacOSX (universal binaries, include i386, x86_64)
Linux (i386, x86_64)

We update the following loaders on request. Please check [our blog](#) and the [loaders page](#) for new versions.

FreeBSD (i386, x86_64)
OpenBSD (i386, x86_64)
Linux ARM (armel)
Linux ARM (armhf) Raspberry Pi (including Pi version 3) and other boards

We are happy to work with our customers if they need bespoke loaders for other custom operating system. Please contact support@sourceguardian.com if you are interested.

We do not include loaders for all supported operating systems to the GUI package. Please visit our [loaders page](#) if you can't find the loader for your OS in the [Copy Loaders](#) dialog.

- We fixed some issues with string concatenation in PHP 5.6 and 7.0 and some other problems. If you experienced issues like that with your code, please re-encode with the latest version 11.1 of SourceGuardian and install the recent loader.

GUI updates

- We fully reworked the files and folders selection dialog which is used when you add files or folder to

the project. This new dialog also uses predefined file filters which are based on your filter settings in File/Preferences.

- We fully reworked files and folders highlighting in the project tree. Folders - bold, virtual folders - green, files/folders changed or added since last encoding - blue.
- Newly added folders and files will be encoded at least once if the "encode only modified" option is selected in advanced settings.
- The option to encode only changed files (in Advanced Settings) always checks modification date for all the files in the project before encoding.
- The project is automatically checked for new and deleted files every time you open the project. This may be turned off/on in [Preferences](#).
- We added samples for code sections in [Advanced Options](#) - code for loader not installed, custom PHP header and custom license text. Try the "Want sample" buttons after clicking "Edit" for the code you want to change.
- We added a new "Copy unencoded" filter section to [Preferences](#). These filters are checked before any further processing but after the exclusion filters. So, files that match any of "copy unencoded" file masks will be copied to the target folder as-is without encoding. E.g. you may now easily encode *.php files but keep *.tpl.php unencoded without manually selecting encoding mode in the project tree.
- If you are locking to a license file, now it's possible to select the folder where the license will be created when you click "Generate License" on the "Lock" screen. Also SourceGuardian will remind you to generate a license file after encoding, this option may be turned off/on in Preferences.
- We added tooltips to the main project window, advanced settings and some features on the lock screen to help our new users.
- You may automatically release the current SourceGuardian license directly from the application. Please find the "Release License" button added to Help/Registration Information. You will be asked for confirmation. Releasing the license lets you reinstall the encoder to another machine or to the same machine after upgrading hardware or OS. If you are going to upgrade the machine or OS, please firstly release the license and then you may transparently re-install your copy of SourceGuardian when the upgrade is complete.

You get 3 free license resets with the initial purchase. If you purchase an additional license or a new copy for another OS, each license also gets 3 resets. If you need to release the license after using all the 3 free resets, please [contact us in support](#).

- Note to Mac users. In this update we are launching the universal GUI also for Mac. It shares the code with Linux and Windows versions and it means it's easier to maintain and it sooner gets the new features and fixes. We will keep the native version of GUI for Mac for some time but it must be considered as discontinued. Both universal and native versions of GUI for Mac will be available for some time and some versions of SourceGuardian. We added support for PHP 7.1 for native GUI for Mac in this release.

Command line encoder updates

- We added automatic registration for command line tools. So, if you install the no-GUI package (Linux)

or get an additional license for using command line tools on another machine, server etc e.g. to generate custom licenses there, you may use automatic registration on the first run. Run sourceguardian command line executable or licgen tool, read and agree with the terms, enter your SourceGuardian online user account email and password when asked in the terminal. If automatic registration can't be used for any reason, e.g. if there is no Internet access, register software as usual. Copy the hex code from the screen, paste it to the user profile, download the encode.lic license file and copy it to the binary folder where the executable is located.

- The command line tools now may be started with GUI license. It's not necessary to specify a path to the GUI encode.lic license file anymore using the -L option. The GUI license will be automatically found if you are starting the command line tools included to GUI installation. However, if you are installing the command line tools separately or to another machine, please register your copy as usual and install the encode.lic license file to the same binary folder where the tools are located or use automatic registration for new copies.
- The license may be released automatically from for the command line tools as well. Please read above as we added the same option to the GUI version. In order to release the license from the command line, use --license-release option with sourceguardian command line tool or licgen and follow the instructions displayed on your terminal.
- We added a new -c (--copy) filter option in addition to -f (--file), -t (--template) and -x (--exclude). The new -c (--copy) option may be used to specify the files that will be copied as-is without encoding to the target folder. This option makes sense only if you specify the target folder with -o (--output) option. If -c is used without -o, then it works as -x and skips the specified files. The new option may take * and ? wildcards or a @filelist.

E.g. you may now encode *.php files but keep *.tpl.php unencoded and copy the latter ones as-is:

```
/path/to/sourceguardian -o /path/to/target -f "*.php" -c "*.tpl.php" /path/to/source
```

- Optional directory trimming level may be specified with -r{n} The default is 0 and means no trimmig, this matches the mode used in previous versions of SourceGuardian. If n is specified, the encoder will remove n folder names from the beginning of target file paths when encoding or copying the files to the target folder. This is similar to -p option of patch utility on Unix. You may find some [samples](#) useful.
- We updated recursive directory search. Source paths containing wildcards in directory names now work, e.g. /path/to/dir??.php This also works in @filelist and you may use it with -f, -t, -c, -x. If the @filelist is specified, recursion is automatically turned on, but it's still possible to change the directory trimming level with -r{n} if necessary.
- {SG_EXPIRY_DATE} tag was added and may be used in the custom text within the generated license files. Use this tag to embed your custom license expiry date into the readable part of the license. Changing of this text in the license file does not make sense and is NOT opening a back door, if you use the expiry date in your custom licenses. The expiry date is stored encrypted as well as other options within the license file. The readable text is only for information and you may put a name of your product, name of your customer and now the expiry date. For further details please see [Advanced Options](#) in GUI or --text option for the [command line licgen tool](#).

9.6 Version 11.0.6 / October 2016

Version 11.0.6 is a minor update release that includes some fixes and updates. We were glad to receive comments and suggestions and want to thank you very much for sharing your ideas! We are looking forward to hearing about other suggestions for improving SourceGuardian.

FIXES

- Fixed `__DIR__` and `__FILE__` constants that were statically compiled when used in initialization expressions in PHP5.6 (older versions of PHP do not allow to use constants in initialization expressions, PHP7 was already handled correctly). If you get issues with using `__DIR__` or `__FILE__` in your PHP 5.6 code, re-encode the files with version 11.0.6 of SourceGuardian and install the updated loaders.
- Fixed `PHP_OS` and some other constants were statically compiled for PHP7. If you get issues with using `PHP_OS` or other constants in your PHP7 code, re-encode with SourceGuardian 11.0.6.

UPDATES

- We updated how empty files are processed. Now empty files are copied to the target folder if it's specified. Empty files are skipped from processing if the target folder is not specified. In either case empty files are not counted as errors anymore. The encoding log will still indicate empty files with [11] code.
- We added the `--verbose` option to the command line encoder. Options are 0-quiet, 1-print only errors in the log, 2-print standard log. 2 is default. You may also add this option to "Additional Command Line Options" in "Advanced settings" if you want to change the encoding log when using GUI.
- The GUI license generator must correctly handle `http://` network paths to the license. In that case the license is created in the target folder under the specified URL path, the `http://` and domain name are filtered to build a correct local path to the license.
- Updated loaders for PHP 5.6 and 7. Please check the [loaders page](#) for new versions.

9.7 Version 11 / June 2016

Version 11 introduces encoding for PHP 7.0 and some new options. As usual this update is partly based on comments and suggestions of our users. We were glad to receive comments and suggestions and want to thank you very much for sharing your ideas! We are looking forward to hearing about other suggestions for improving SourceGuardian. Here is a list version 11 changes.

NEW FEATURES

- Full support of PHP 7.0 encoding including the latest language options: scalar type declarations, return type declarations, new operators, anonymous classes and more.

Encoding for PHP 7.0 is fully supporting in version 11 of SourceGuardian. PHP 7.0 introduced new language features and updated bytecode format to support them. Files encoded with SourceGuardian 10.x or older need to be re-encoded with SourceGuardian 11 in order to run protected files under PHP 7.0

- New loaders for PHP 7.0, updated loaders for older versions of PHP

We updated loaders and created new ones for PHP 7.0. Loaders for the following operating systems are available:

Windows 32-bit (VC6, VC9; VC11 PHP 5.5, 5.6; VC14 PHP 7.0)
Windows 64-bit (VC9 PHP 5.3,5.4; VC11 PHP 5.5,5.6; VC14 PHP7.0)
MacOSX (universal binaries, include i386, x86_64)
Linux (i386, x86_64)

We update the following loaders on request. Please check [our blog](#) and the [loaders page](#) for new versions.

FreeBSD (i386, x86_64)
OpenBSD (i386, x86_64)
Linux ARM (armel)
Linux ARM (armhf) Raspberry Pi (including Pi version 3) and other boards

We are happy to work with our customers if they need bespoke loaders for other custom operating system. Please contact support@sourceguardian.com if you are interested.

We do not include loaders for all supported operating systems to the GUI package. Please visit our [loaders page](#) if you can't find the loader for your OS in the [Copy Loaders](#) dialog.

- We removed the @SourceGuardian tag from all the encoded files excepting files encoded with so-called "[conjunction](#)" option.
- Locking to a domain automatically enables www. subdomain and you do not need to specify the www. subdomain on the locking page.
- The command line tools now return expected exit codes. The encoder returns encoding status after processing a single file. When it is processing multiple files it returns zero in case of running the process and if there are no issues in using the command line options and then you need to check the encoding log for further details. Licgen returns exit code on invalid options or status of the license generation. Sginfo returns exit code on invalid options or status of the encoded file. Please find further details in the [Exit codes](#) section.
- A built-in file information tool or its command line equivalent now returns estimated memory usage for running the encoded file. This does not include any additional memory requirements of your code. Estimated memory usage also differs for versions of PHP as lengths of bytecode may vary.
- A new `sg_loader_version()` function returns version of the loader. As well as any other loader functions it's available only when the loader is loaded into PHP, i.e. you may use it from your protected files or if the loader is installed to `extension_dir` and `php.ini`.

9.8 Version 10 / June 2014

Version 10 introduces improved code protection methods as well as encoding for PHP 5.6 and some new options. As usual this update is partly based on comments and suggestions of our users. We were glad to receive comments and suggestions and want to thank you very much for sharing your ideas! We

are looking forward to hearing about other suggestions for improving SourceGuardian. Here is a list of recent version 10 changes.

NEW FEATURES

- Improved code protection methods
- Full support of PHP 5.6 encoding including the latest language options: constant scalar expressions, variadic functions operator, updated *use* operator and more

Encoding for PHP 5.6 is fully supporting in version 10 of SourceGuardian. PHP 5.6 introduced new language features and updated bytecode format to support them. Files encoded with SourceGuardian 8, 9.x or older will need to be re-encoded with SourceGuardian 10 in order to run protected files under PHP 5.6

- New loaders for PHP 5.6, updated loaders for older versions of PHP

We updated loaders and created new ones for PHP 5.6. Loaders for the following operating systems are available:

- Windows 32-bit (VC6, VC9; VC11 PHP 5.5, 5.6)
- Windows 64-bit (VC9 PHP 5.3,5.4; VC11 PHP 5.5,5.6)
- MacOSX (universal binaries, include i386, x86_64)
- Linux (i386, x86_64)
- FreeBSD (i386, x86_64)

We update the following loaders on request. Please check [our blog](#) and the [loaders page](#) for new versions.

- OpenBSD (i386, x86_64)
- IBM PowerLinux
- HP-UX Itanium
- Linux ARM (armel)
- Linux ARM (armhf) including Raspberry Pi

We are happy to work with our customers if they need bespoke loaders for other custom operating system. Please contact support@sourceguardian.com if you are interested.

We do not include loaders for all supported operating systems to the GUI package. Please visit our [loaders page](#) if you can't find the loader for your OS in the [Copy Loaders](#) dialog.

- A closing PHP tag is not added anymore to the end of encoded files. It's not needed, Zend recommends to not use it. Omitting the closing tag 'automatically' protects against 'headers already sent' errors if any of the encoded files were accidentally opened/saved in the editor and some characters were added at the end. Omitting the closing tag does not affect execution of protected files.
- SG_LIC_PATH environment variable may be used to specify where the loader should search for a license file. See details for [GUI](#), for [command line](#)

- PHP short tags <? ?> are enabled by default. If you do not need them enabled for any reason, you may turn them off in [advanced settings in GUI](#) or by using the new [--no-short-tags option in the command line](#). The old --short-tags option has been removed.
- License generation fixed in GUI when a URL is specified as a license file name.
- Fixed how encoding only of changed files works in GUI.
- A new 'Refresh' button added to the GUI for updating the project tree. It is useful if any of the files were changed or added to the folders behind the encoder GUI. Also automatic refresh happens when you encode only files updated since the last encoding (see [Advanced settings](#)).
- A new --keep-file-date [command line option](#) added to keep the modification date for encoded files the same as the modification date of source files. The same option is available in GUI in [Advanced settings](#).

9.9 Version 9.5 / July 2013

Version 9.5 introduces encoding for PHP 5.5 and adds some new options. As usual this update is partly based on comments and suggestions of our users. We were glad to receive comments and suggestions and want to thank you very much for sharing your ideas! We are looking forward to hearing about other suggestions for improving SourceGuardian. Here is a list of recent version 9.5 changes.

NEW FEATURES

- Full support of PHP 5.5 encoding including the latest language options: generators, coroutines, 'finally' operator and more

Encoding for PHP 5.5 is fully supporting in version 9.5 of SourceGuardian. PHP 5.5 introduced new language features and updated bytecode format to support them. Files encoded with SourceGuardian 8, 9.0 or older will need to be re-encoded with SourceGuardian 9.5 in order to run protected files under PHP 5.5

- New loaders for PHP 5.5, updated loaders for older versions of PHP

We updated loaders and created new ones for PHP 5.5. Loaders for the following operating systems are available:

- Windows 32-bit
- Windows 64-bit (PHP 5.3,5.4,5.5)
- MacOSX (universal binaries, include i386, x86_64)
- Linux (i386, x86_64)
- FreeBSD (i386, x86_64)
- OpenBSD (i386, x86_64)
- IBM PowerLinux
- HP-UX Itanium
- Linux ARM (armel)

Linux ARM (armhf) including Raspberry Pi

We are happy to work with our customers if they need bespoke loaders for other custom operating system. Please contact support@sourceguardian.com if you are interested.

We do not include loaders for all supported operating systems to the GUI package. Please visit our [loaders page](#) if you can't find the loader for your OS in the [Copy Loaders](#) dialog.

- Standard input and output support for the [command line encoder](#).
- New option to disable additional CRC check to enable eval()'ing protected code. See details for [GUI](#), for [command line](#)
- Added "Script will expire in (days)" option to the [Locking page](#).
- Changed how 'modification date' works in GUI
- The 'Copy unencoded' mode is now assigned to empty files when adding them to the project. This is to eliminate 'empty file - skipped' error messages from the encoder. It replicates what GUI does with other files that are not detected by their file extensions. If you don't need to copy empty files to the target folder, you may change the encoding mode to 'Skip' for them.
- Reverting the project was affecting modification dates - fixed
- Updated built-in support and automatic update in GUI.

9.10 Version 9.0 / July 2012

Version 9.0 introduces encoding for PHP 5.4, fixes problems and adds some new options. The update is partly based on comments and suggestions of our users. We were glad to receive comments and suggestions and want to thank you very much for sharing your ideas. We are looking forward to hearing about other suggestions for improving SourceGuardian and we are open to new ideas. Here is a list of recent version 9.0 changes.

NEW FEATURES

- Full support of encoding for PHP 5.4

Encoding for PHP 5.4 is fully supporting in version 9 of SourceGuardian. PHP 5.4 introduced new language features and updated bytecode format to support them. Files encoded with version 8 or older of SourceGuardian will need to be re-encoded with SourceGuardian 9 in order to run protected files in PHP 5.4

- New loaders for PHP 5.4, updated loaders for older versions of PHP

We updated loaders and created new ones for PHP 5.4. Loaders for the following operating systems are available:

Windows 32-bit
Windows 64-bit (PHP 5.3,5.4)
MacOSX (universal binaries, include i386, ppc, x86_64, ppc64)
Linux (i386, x86_64)
FreeBSD (i386, x86_64)
OpenBSD 5.0,5.1 (i386, x86_64)

We may try to compile bespoke loader for other custom operating system. Please contact support@sourceguardian.com if you are interested.

- New option to stop encoding on E_DEPRECATED PHP 5.3+ compiler errors. See details for [GUI](#), for [command line](#)
- New option to encode only changed files detected by file modification date. See details for [GUI](#), for [command line](#)
- Allow absolute file system path to a custom license file or an URL. [See details](#)
- Improved compatibility for scripts encoded for PHP 5.2.x
- Automatic filtering UTF-8 BOM from all source files. This allows encoding files created in editors that save BOM.
- Protected scripts starter code has been reworked and improved. The default starter code now recognizes "thread safety" option providing a correct loader name for systems running thread safe PHP. It also displays instructions to a user about what loader is needed, where to download it and how to install it. The old version was displaying the loader name and the error message. It is still possible to exclude the default starter code from protected scripts (using -n command line encoder option or an appropriate option in GUI). Also it is still possible to replace the standard error message about a missing loader (using -j command line option or an appropriate option in GUI) with a custom one which may be either html/text or php code. If it's PHP code a new `$_ixedurl` variable may be used, it contains the required loader download link.

For information: The starter code is a non-encoded portion of php code that is added by default to encoded PHP scripts. The task of the starter code is to detect if SourceGuardian loader is present and if it is not, then try to load it automatically if PHP configuration allows to do it. If automatic loading is not possible, the starter code will display a message telling a user what loader is needed, where to download it and how to install it in order to run SourceGuardian protected files.

- Now it is possible to send licgen's output to console instead of a file, use -- (double dash) instead of the output file name. [See details](#)
- Custom constants substitution in the custom header code. All user {constants} will be replaced in the prepend code. Also some standard SG constants may be used:

{SG_DATE} - current date i.e. date of encoding
{SG_LICENSEE} - SG license owner from the SG license file
It works in the same way also for licgen and do replacements for custom text if it is used.

BUGFIXES

- Fixed displaying strict errors (`--strict-errors`) that did not work if displaying deprecated errors (`--deprec-errors`) was not specified.
- On multiple compiling error the encoder was displaying the last one, while PHP displays the first one. Fixed.
- Updated protected scripts header. A protected script will not stop execution in case of 'extension_dir does not exists' error, the script will continue and fails in a standard way or an assigned error handler function will be called. This allows to catch errors like that in the custom error handler.
- Fixed encoding of PHP predefined constants like `DIRECTORY_SEPARATOR` in PHP 5.3+ (re-encoding is required). Protected scripts always do dynamic lookup as values of standard constants may differ on systems depending on OS, CPU etc.
- Fixed running domain locked scripts on web servers returning a port number in `HTTP_HOST/ SERVER_NAME` (e.g. `mydomain.com:88`)
- Fixed APC and other PHP bytecode cache compatibility issue in loaders.
- Now it is possible to catch SourceGuardian errors like "SourceGuardian Loader - script header is broken [10]" etc with a standard PHP error handling mechanism.
- Fixed `--short-tags` option which was always on in the previous versions of the encoder.

MacOSX GUI

- Two Project Window modes: "standard" and "file manager". This may be changed in Preferences, please read further details in Preferences / Interface & Updates.
- More informative log. Added total files counter, copied files counter and list, additional warning if the encoder has not completed encoding successfully.
- Exclusions in Preferences was changed to file masks instead of file extensions.
- Project window. "Don't encode" type renamed to "Copy unencoded", added new "Skip" type which are more straightforward.
- Support of the new options of version 9 encoder (PHP 5.4 encoding, new loaders, new advanced options etc)

NEW GUI FOR WINDOWS AND LINUX

- We are proud to present a fully reworked new cross platform GUI for Windows and Linux. It includes all the features of GUI for OSX and even more. It has built-in support system and more.

SUPPORTED PHP VERSIONS

- Encoding for PHP 4.3.x to PHP 5.4.x are fully supported

SUPPORTED OS

- Encoder is available for Windows, MacOSX, Linux (i386 and x86_64 versions).
- GUI and command line encoders and tools are included.

9.11 Version 8.2 / April 2010

Version 8.2 is partly based on comments and suggestions of our users. We were glad to receive them and want to thank you very much for sharing your ideas. We are looking forward to hearing about other suggestions for improving SourceGuardian and we are open to new ideas. Here is a list of recent version 8.2 changes.

- **Added a new predefined loader constant 'loader_version'**

`sg_get_const('loader_version')` returns a loader version number when called from the protected code.

- **Added a new license generator option to add custom text into the license file.**

We have added an option to include custom text into license files generated by SourceGuardian license generator. The included text is protected with a checksum against modification. The option may be used to include user information, license description etc into license files.

- **API function `sg_get_const()` issue fixed.**

In previous versions it was possible to get an encoded constant value from the unencoded code included from the protected one. It has been fixed.

- **The last catch block issue fixed.**

The last catch block did not terminate the execution of the code in previous versions. If you are experiencing this problem you need to re-encode your code with version 8.2 of the encoder and install the updated loader.

- **Mac GUI improvements.**

We have added an exclusion list to Preferences which lets exclude files or folders that should not be included into the project when you add files. You can easily exclude `.svn` folders for example.

Added an option to encode only selected folders or files - you need to press Command key while clicking the Encode button. This option is useful if you need to re-encode only some files without doing it for the entire project.

Added Custom license text in Advanced settings which reflects a new generator option mentioned above.

New Preferences screen.

Other minor fixes.

- **Documentation update.**

The loaders installation section has been revised, added important information for Windows users. Other

minor changes have been done reflecting the improvements mentioned above.

9.12 Version 8.1 / January 2010

Version 8.1 is an update based mainly on comments and suggestions of our users. We were glad to receive them and want to thank you very much for sharing your ideas. We are looking forward to hearing about other suggestions for improving SourceGuardian and we are open to new ideas. Here is a list of recent version 8.1 changes.

- **New loader API functions for encryption and decryption of a string.**

We have added new loader API functions `sg_encode_string()` and `sg_decode_string()` which can be used for encryption and decryption of a string when called from the protected PHP code. See [SourceGuardian loader API](#) section for details.

- **New algorithm for searching for the loader**

We have changed the way protected scripts search for loaders when dynamic loading is possible. Now protected scripts use the following rules:

<u>Version</u>	<u>Logic</u>
PHP 5.2.5+	Loader is loaded? If no, try <code>extension_dir</code> , then custom error or standard message.
<PHP 5.2.5	Loader is loaded? If no, try to <code>dl()</code> from the current dir then parents, if not found - try <code>extension_dir</code> . If still not found, then custom error or standard message.

Note: dynamic loading is only possible from `extension_dir` since PHP 5.2.5 and we do not try any other directories for PHP 5.2.5+

- **Option to stop encoding at a critical error**

We have added a new option for the encoder to stop at a critical error. See [GUI Manual / Project / Advanced Options](#) and [Using the command line encoder / Advanced Options](#) sections for details.

- **Option to report E_STRICT compiler errors**

E_STRICT "Strict Standards" warnings were introduced in PHP 5. Now the encoder has an option to warn of such messages during encoding. See [GUI Manual / Project / Advanced Options](#) and [Using the command line encoder / Advanced Options](#) sections for details.

- **User files encoding API changes**

We have changed the way how script keys based on Project ID or Project Key (if an external license file is used) are used for encoding or decoding files with `sg_load_file()` and `sg_encode_file()` loader API functions. Now we keep track of which protected PHP script uses which key and the loader will use an appropriate key for encryption/decryption of user files and strings (see new `sg_encode_string()` and

sg_decode_string() API functions). Older versions of loaders would use the key of the last loaded protected script. These changes allow to use the encoding functions correctly in a situation when one protected script is inherited from another protected script and the above encoding/decoding functions are called. See [Creating custom encoded files from protected scripts](#) section for details of using these functions.

- **Marking a file to be skipped during encoding**

It's possible to mark a file to be skipped by the encoder. Add the following string anywhere in the code, use comments. Skipped files will be copied as-is to the target folder if it's specified.

```
SourceGuardian:DO_NOT_ENCODE
```

E.g. `/* SourceGuardian:DO_NOT_ENCODE */`

Note: Comparison is case sensitive for Windows and Solaris if you will ever move to any of these platforms for encoding your source files. Do not change the case for better code compatibility.

- **IPv4 address mapped to IPv6 is correctly recognized**

The loaders now correctly understand IPv4 mapped to IPv6 `::ffff:XX:XX:XX:XX` style IP addresses in `$_SERVER['SERVER_ADDR']`. This fixes issues with locking to IP when a webserver returns IPv6 style address.

Note: SourceGuardian does not support locking to IPv6 yet, only to IPv4 addresses.

- **Tested compatibility with xdebug extension**

We have tested compatibility of loaders and xdebug PHP debugging extension. There is NO any compatibility issue except for the scripts encoded with `--conj` (script will only work with other encoded files) option. As xdebug modifies the bytecode files encoded with this option will fail with a conjunction error (code [15]). It's NORMAL behaviour demonstrating one of the points of protection. Obviously the internals of the encoded script cannot be debugged so once the debugger reaches the `sg_load()` function the entire protected script will be executed in one step.

- **Tested compatibility with APC**

We have tested compatibility of loaders and APC PHP cache extension. APC v3.0.19 the latest stable was tested. All protected scripts get cached no problem.

- **Automatic update of the project's contents for macOS GUI**

The contents of a project (files and folders added to the project) will be automatically synchronized with the contents of real folders on a disk when the project is reopened in SourceGuardian. If there are any changes within the added folders then the inner list of files and folders will be recursively updated. If there are any changes of the folders or files which were directly added on the top level then the application will query whether to remove missed elements or leave them within the project.

Note: Newly added files will get an encoding type according to their file extensions and settings in preferences. Probably you will want to check the type setting for new files.