ionCube Loader 5.0 User Guide

This document describes the available php.ini configuration options of the ionCube Loader that relate to processing of PHP encoded files, and also the ionCube24 service.

ENCODED FILES

INI entry: `ioncube.loader.encoded_paths`

Purpose: Specify the locations of encoded files

The ionCube Loader will normally examine a PHP file before processing to test whether it is encoded, and will run the file itself if necessary. Although this checking is very efficient, restricting which files the Loader tests for being encoded may give extra performance. If set to a series of paths or files, only files in those locations are tested.

Entries should be separated by a `:` on Unix and `;` on Windows. A path may be prefixed with `+` or `-` to add or remove that path from the possible locations. `+` is assumed if no character is given.

Examples: (`...` means `ioncube.loader.encoded_paths`)

- Site with a single encoded module in `/var/www/html/modules/RSS`

```
... = "/var/www/html/modules/RSS"
```

- As above, with a site configuration file encoded too.

```
... = "/var/www/html/modules/RSS:/var/www/html/config/config.php"
```

- Encoded files may be anywhere except for `/var/www/html/framework`

```
... = "/:-/var/www/html/framework"
```

- Site with most modules encoded except for one

```
... = "/var/www/html/modules:-/var/www/html/modules/plain"
```

- As above, with an encoded config file in the plain directory

```
... = "/site/modules:-/site/modules/plain:/site/modules/plain/config.php"
```

Locations:

The `ioncube.loader.encoded_paths` property can be set in a `php.ini` file, in a `.htaccess` file (when using Apache), in a `.user.ini` file (when using CGI PHP 5.3+) or using `ini_set` within a PHP script. In ini files only the last value will be used for the encoded_paths property. If you wish to build up the value in several lines then, for PHP 5.1+, you can use the following syntax:

```
ioncube.loader.encoded_paths = "/path1"
ioncube.loader.encoded_paths = ${ioncube.loader.encoded_paths}":/path2"
```
; etc...

IONCUBE24 : real-time intrusion protection

(Available for Linux 32 and 64 bit servers)

ionCube24 (https://ioncube24.com) is a new ionCube service that offers real-time protection against the exploit of vulnerabilities leading to PHP code to be modified or inserted into a website. Such vulnerabilities are widespread and often result in website defacement or customer data being compromised. Our weekly security roundup on our blog (http://blog.ioncube.com) regularly features new vulnerabilities, with plugin based systems such as Wordpress frequently having issues.

You can sign up and use the service for free, and the server side support is built into the Linux Loaders as standard. With the Loader installed, ionCube24 can be activated at any time to give active intrusion protection.

Performance may also be improved with ionCube24 enabled, particularly coupled with the PHP Op Cache in PHP 5.5+, as the Loader learns and remembers which files are encoded, eliminating any overhead for regular files.

php.ini settings

There are a number of ini settings, summarised below. The setup process at https://ioncube24.com automatically gives the settings that you need to get started, but you may wish to make changes yourself once setup. The default values are given with each example.

INI entry: `ic24.enable = 0 ; default off`

Purpose: Enable or disable ionCube24 features. This defaults to off, and in this case no ionCube24 behaviour is activated.

Example:

```
ic24.enable = 1
```

INI entry: `ic24.api_access_key ; provided during setup`

Purpose: A key that is part of the authentication process for adminstration requests. This value is provided when adding a server to ionCube24.

INI entry: `ic24.api_check_ip = 1 ; default on`

Purpose: If set, ionCube24 refuses access to API functions unless the calling IP is a known ionCube IP address. This option should be left with the default setting unless web requests pass through a proxy and your site is accessed from the IP of the proxy instead of the client. Note that access to API functions will still be authenticated by access key.

INI entry: `ic24.sec.enable = 1 ; default on if ic24.enable is on`

Purpose: Enable the intrusion protection part of ionCube24. This defaults to on, but will automatically be off if the global `ic24.enable` has not been set.

INI entry: `ic24.sec.exclusion_key ; provided during setup`

Purpose: A key that if present at the start of a file, will identify the file as trusted. This value is provided when adding a server to ionCube24.

INI entry: `ic24.home_dir ; no default`

Purpose: The home directory for ionCube24 related system files. A location outside of the web root is recommended. It should be writable by the web server during startup.

Example:

```
ic24.home_dir = /var/www/ic24_home
```

INI entry: `ic24.sec.trusted_include_paths ; no default`

Purpose: List paths from where files can be included and automatically trusted.

Example:

```
ic24.sec.trusted_include_paths = "/var/cache:/var/cache2"
```

Directories can be excluded from the list by prefixing with a minus character -. e.g.

```
"/var/cache:-/var/cache/subdir"
```

This is useful if your site creates and/or modifies files by itself from time to time, e.g. in a cache

directory, though we would recommend producing files that include the exclusion key as an alternative. Requests that *directly* access files from a trusted include path will be blocked but the file itself will not be blocked, so requests that use the file as intended will still work. See ioncube24.com for more details once signed up.

INI entry: `ic24.sec.block_uploaded_files = 1 ; default on`

Purpose: If set, ionCube24 blocks from execution any uploaded files that are processed using the standard PHP mechanism for uploaded files. This applies even if the file is subsequently included and where included files being automatically approved with the previous setting.

INI entry: `ic24.sec.block_stdin = 1 ; default on`

Purpose: Refuse code that PHP sees via `stdin`. If disabled, code via `stdin` will run without security checking as there is no filepath. This setting should be left on as PHP would normally never receive a script via `stdin`.

INI entry: `ic24.update_domains_retry_interval = 5 ; default 30`

Purpose: If fetching the set of domains being managed fails, retry after the specified number of seconds.
