

---

---

# **Guía del usuario**

---

---

## ***¿Qué es BlackSEO Detector?***

Bueno... Esta pregunta no es tan fácil de responder.

Por un lado, “BlackSEO Detector” es en realidad una clase escrita en Ruby. Por otro, se ha realizado una prueba de concepto utilizando esta clase y su nombre también es “BlackSEO Detector”.

En este documento, el nombre “BlackSEO Detector” se usará para referirse a esta prueba de concepto.

## ***¿Qué hace BlackSEO Detector?***

BlackSEO Detector usa buscadores como Bing o Google para buscar contenidos extraños en webs. Estos contenidos pueden ser indicadores de intrusiones en los servidores o en las infraestructuras utilizadas.

Después, analiza las respuestas de los servidores bajo una serie de condiciones. Los resultados de estas pruebas pueden ser usados para tratar de determinar si existe realmente problemas de seguridad.

Con toda la información recolectadas en este proceso se crea un informe.

## ***Características***

BlackSEO Detector funciona en modo batch. Por defecto, no proporciona ninguna salida de texto en consola. De este modo puede ser utilizado para programar tareas.

Toda la configuración de BlackSEO Detector se realiza mediante un fichero de configuración. Los detalles sobre su formato pueden consultarse en la sección “Fichero de Configuración”.

BlackSEO Detector genera un informe HTML con la información obtenida.

BlackSEO Detector tiene las siguientes características:

- **Soporte para motores de búsqueda:** En la actualidad están soportados Google Custom Search API and Bing Search API. Los usuarios pueden añadir sus implementaciones de otros motores de búsqueda.
- **Múltiples consultas:** Se puede combinar varias consultas a motores de búsqueda en un proyecto.
- **Soporte de idiomas:** Se incluyen los ficheros de traducción para español e inglés. Los usuarios pueden añadir a la aplicación sus propios ficheros de traducción.
- **Soporte para proxy.** Las conexiones pueden establecerse de forma directa, a

través de proxy o utilizando la configuración indicada en la variable de entorno `http_proxy` (sólo en Ruby 2 o superior).

- **E-mail reporting:** Los informes generados pueden ser enviados automáticamente por correo electrónico.

## Uso

El fichero principal de BlackSEO Detector es `poc.rb`. Sólo admite un argumento de línea de comandos: la ruta del fichero de configuración. De este modo puede invocarse mediante:

```
ruby poc.rb ruta_para_el_fichero_de_configuracion/config.txt
```

...o, si el sistema operativo esta configurado para ejecutar directamente los ficheros de Ruby, basta con

```
poc.rb ruta_para_el_fichero_de_configuracion/config.txt
```

## Fichero de Configuración

Los detalles sobre el formato del fichero de configuración de proyecto están disponibles en la sección “Formato del fichero de configuración” de la “Clase `My_configuration`”, en la documentación de las “Componentes Software”

Las opciones de configuración son:

NAME	TYPE	DEEFAU LT VALUE	COMMENTS
verbose	Boolean	N	Si vale true, la aplicación mostrará mensajes en tiempo real acerca de las actividades que realiza. <b>Ejemplo:</b> verbose=Y
language	String	spanish	Idioma para el informe <b>Ejemplo:</b> language=english
search_engine	Pairs  mandatory		Asigna nombres para referenciar los motores de búsqueda. El primer elemento del par es el nombre del motor de búsqueda . El segundo elemento es el nombre de la clase Ruby que implementa el

			<p>motor de búsqueda</p> <p><b>Ejemplo:</b> search_engine=!Google!Google_cse</p>
search_engine_option	Pairs		<p>Permite configurar la inicialización de los objetos que implementan los motores de búsqueda.</p> <p>El primer elemento del par es el nombre del motor de búsqueda.</p> <p>El segundo elemento es el valor del parámetro de inicialización.</p> <p>Los valores serán pasados al método de inicialización del objeto en el mismo orden en que se proporcionan en el fichero de configuración. Para más detalles, consúltase “Clase Search_engine”, “Clase Goolge_cse” y “Clase Bing_api”</p> <p><b>Ejemplo:</b> search_engine_option=!Google!abcd search_engine_option=!Google!efgh</p>
title	String	Report	<p>Título principal del informe a generar</p> <p><b>Ejemplo:</b> title=BlackSEO Detector test</p>
report_file	String mandatory		<p>Ruta para el fichero de configuración.</p> <p>Si se trata de una ruta relativa, se tomará como base el directorio en que se encuentre el fichero de configuración.</p> <p><b>Ejemplo:</b> report_file=reports/report.html</p>
add_date_time	Boolean	Y	<p>If true, the date and time when the test started will be appended to the report file name.</p> <p><b>Ejemplo:</b> add_date_time=N</p>
overwrite_report	Boolean	N	<p>Si vale true, el fichero será sobrescrito en caso de existir.</p> <p><b>Ejemplo:</b> overwrite_report=Y</p>
proxy	String	:ENV	<p>Servidor proxy.</p> <p><b>Ejemplo:</b> proxy=localhost</p>
proxy_port	String		Puerto del proxy

			<b>Ejemplo:</b> proxy_port=80
proxy_user	String		Usuario para autenticación del proxy <b>Ejemplo:</b> proxy_user=username
proxy_password	String		Contraseña para autenticación del proxy <b>Ejemplo:</b> proxy_password=password
user_agent	String	BlackSeo Detector	User Agent usado por la aplicación para realizar peticiones HTTP <b>Ejemplo:</b> user_agent=Custom UA
max_redirects	Integer	5	Número máximo de redirecciones a seguir al realizar una petición HTTP <b>Ejemplo:</b> max_redirects=2
hard_check	Boolean	N	Si es falso, los tests para una URL finalizarán tan pronto como un test revele la existencia de problemas. En otro caso, todos los tests serán realizados para todas las URLs <b>Ejemplo:</b> hard_check=Y
max_report	Integer	25	Número máximo de problemas a investigar en un informe. Una vez alcanzado, las URLs restantes no serán descargadas <b>Ejemplo:</b> max_report=10
max_host	Integer	15	Número máximo de problemas a investigar para un host. Una vez alcanzado, sus URLs restantes no serán descargadas <b>Ejemplo:</b> max_host=10
max_folder	Integer	5	Número máximo de problemas a investigar para un directorio. Una vez alcanzado, sus URLs restantes no serán descargadas <b>Ejemplo:</b> max_folder=1

send_mail	Boolean	N	Si es true, el informe generado sera enviado por correo <b>Ejemplo:</b> send_mail=Y
mail_server	String		Servidor de correo <b>Ejemplo:</b> mail_server=localhost
mail_port	Integer	:default	Puerto del servidor de correo. <b>Ejemplo:</b> mail_port=123456
mail_cypher	String	tls	Tipo de cifrado para el envío de correos. Valores validos: none, tls, ssl <b>Ejemplo:</b> mail_cypher=none
mail_user	String		Cuenta de usuario para autenticar el envío de correos. <b>Ejemplo:</b> mail_user=mailuser
mail_password	String		Contraseña para autenticar el envío de correos. <b>Ejemplo:</b> mail_password=mailpasswd
mail_auth	String		Tipo de autenticación para el envío de correos. Ver la documentación de Net::SMTP documentation para más detalles <b>Ejemplo:</b> mail_auth=login
mail_from	String		Remitente de correo <b>Ejemplo:</b> mail_from=sender@example.com
mail_to	String		Destinatario de correo <b>Ejemplo:</b> mail_to=admin@example.com
mail_subject	String	BlackSeo Detector Results	Asunto a indicar en los mensajes <b>Ejemplo:</b> mail_subject=Test Results
mail_body	Multiple strings		Cuerpo de los mensajes. Las líneas serán dispuestas en el cuerpo del mensaje en el orden en que se proporcionen en el fichero de configuración.

			<p>Se realizarán las siguientes sustituciones de textp:</p> <ul style="list-style-type: none"> <li>- Las líneas que tengan sólo un punto (.) serán sustituidas por líneas vacías.</li> <li>- [[problems_found]] se reemplazará por el número de problemas encontrados</li> <li>- [[report_file]] se reemplazará por la ruta del fichero de informe</li> </ul> <p><b>Ejemplo:</b>  mail_body=Task completed.  mail_body=.  mail_body=See [[report_file]].</p>
Query	Pairs mandatory		<p>Consultas a realizar a los buscadores.</p> <p>El primer elemento del par es el nombre del motor de búsqueda.</p> <p>El segundo, la búsqueda a realizar</p> <p><b>Ejemplo:</b>  query=!Google!levitra site:example.com</p>
allowed_domain	Multiple strings		<p>Dominios a incluir en el reporte.</p> <p>Las URLs de dominios no incluidos en esta lista serán ignoradas.</p> <p>Las redirecciones a dominios no incluidos en esta lista producirán una alerta en el informe.</p> <p><b>Ejemplo:</b>  allowed_domain=example.com  allowed_domain=example.net</p>
ignore_files	Multiple strings		<p>Expresiones regulares. La URLs que se ajusten a estos patrones serán ignoradas.</p> <p><b>Ejemplo:</b>  ignore_files=^.*generate_pdf=</p>
text	Multiple strings		<p>Textos usados como indicadores de problemas.</p> <p>Cuando uno de estos textos es encontrado en respuestas, URLs o sumarios, se genera una alerta en el informe</p> <p><b>Ejemplo:</b>  text=levitra  text=viagra</p>

## ***Ejemplo de fichero de configuración***

```
# Langage
language=english

# Verbose mode
verbose=y

# Search Engines
search_engine==Google - CSE=Google_CSE
search_engine_option==Google - CSE=xxx_api_key_xxxxx
search_engine_option==Google - CSE=yyy_engine_id_yyyy

search_engine==Bing - API=Bing_api
search_engine_option==Bing - API=zzz_api_key_zzzz

# Report title
title=Report for EXAMPLE.COM

# Report File
report_file=reports/example.com.html

# Must add date and time to file name?
add_date_time=n

# Overwrite report file if already exists
overwrite_report=y

# Keep looking for problems even if one has already been detected for the URL?
hard_check=y

# User agent to use in web access
user_agent=Mozilla/5.0 (Windows NT 6.1; WOW64; rv:32.0) Gecko/20100101 Firefox/32.0

# Proxy information. :ENV is used if no configuration is given
proxy=
proxy_port=
proxy_user=
proxy_password=

# Queries to make .There may be more than one 'query' line"
```



```
query==Google - CSE=site:example.com levitra online
query==Bing - API=site:example.com levitra online

# Text pointing to problems. There may be more than one 'text' line
text=viagra
text=levitra

# Maximum number of redirections to follow for each URL
max_redirects=5

# Domain allowed for reporting and redirection.
allowed_domain=example.com
allowed_domain=example.net

# Regular expression. URLs that match this regexp will not be analysed
ignore_files=.*generate_pdf

# Maximum number of files to download per report, host and folder
max_report=25
max_host=20
max_folder=10

# Whether or not to send email with results
send_mail=y

# Mail configuration
mail_server=mail.example.com
mail_port=
mail_cypher=tls
mail_user=mailusername
mail_password=mailpassword
mail_auth=login
mail_from=mailusername@example.com
mail_to=mailusername@example.com
mail_subject=BlackSEO Detector Report for EXAMPLE.COM
mail_body=A BlackSEO Detector report for EXAMPLE.COM has been generated
mail_body=.
mail_body=Problems found: [[problems_found]] (se attached file)
mail_body=.
mail_body=Full listing is located at [[report_file]]
```

---

---

## **Añadir funcionalidades**

---

---

## ***Añadir soporte para motores de búsqueda***

Para añadir soporte para un motor de búsqueda es necesario definir una clase de Ruby.

Esta clase debe implementar todos los métodos públicos definidos en la “Clase Search\_engine”. En el directorio “searchengines” de la aplicación hay ejemplos de implementaciones.

El unico método cuyos parámetros pueden diferir de los de la clase Search\_engine es initialize.

Esta clase debe estar incluida en un fichero de código Ruby con extensión “rb”. El nombre del fichero debe ser el mismo de la clase.

Este fichero debe estar localizado en el directorio “searchengines” de la aplicación.

## ***Añadir soporte para idiomas***

Para añadir soporte para un idioma debe crearse un fichero de traducción en el directorio “lang” de la aplicación BlackSEO Detector.

La extensión del fichero debe ser “txt”. El nombre del fichero debe ser el nombre del idioma. Por ejemplo: spanish.txt

El fichero debe seguir las especificaciones de “Formato de fichero de tabla de traducción” y traducir las siguientes definiciones:

<b>analysis_omitted_protocol:</b> Analysis omitted for URL with non supported protocol - [[protocol]]
<b>analysis_omitted_regexp:</b> Analysis omitted. URL matches exclusion regexp
<b>analysis_omitted_type:</b> Analysis omitted for [[document_type]] file
<b>authorized_domain:</b> Authorized domain
<b>cache_from:</b> CACHE FROM
<b>cannot_download:</b> Could not download the resource
<b>data:</b> Data
<b>desination_page_problems:</b> Destination page may have security problems. Please check
<b>destination_domain_problems:</b> Destination domain may have security problems. Please check
<b>details_for:</b> Details for
<b>dns_compromise:</b> This may be an indicator of DNS server compromise
<b>external_redirection:</b> EXTERNAL REDIRECTION
<b>file:</b> FILE
<b>fit_to_width:</b> Fit to width

**folder:**FOLDER  
**forging\_referer\_of:**FORGING REFERER OF  
**found:**FOUND  
**generated\_by:**Report generated by BlackSEO Detector  
**hash\_contains\_text:**URL hash string contains the text "[[text]]"  
**hide\_data:**Hide data  
**hide\_show\_analysis:**Hide / Show analysis data  
**host:**HOST  
**host\_name\_contains\_text:**Host name "[[host]]" contains the text "[[text]]"  
**html:**HTML  
**html\_response:**HTML Response  
**http:**HTTP  
**http\_headers:**HTTP Headers  
**info:**Info  
**last\_if\_redirs:**Last ones if redirections were followed  
**malicious\_redirection:**Malicious redirection?  
**max\_problems\_folder:**Maximum number of problems investigated for folder met  
**max\_problems\_host:**Maximum number of problems investigated for host met  
**max\_problems\_report:**Maximum number of problems investigated in report met  
**more\_details:**Click here for more details  
**no\_files\_downloaded:**No files were downloaded  
**no\_mailicious\_content:**Could not find malicious content. Anyway, still there might be issues.  
**no\_problems:**No problems found  
**no\_problems\_in\_url:**No problems found in URL  
**no\_redirection:**NO EXTERNAL REDIRECTION  
**no\_summary:**No result summary to show  
**no\_url\_to\_translate:**Cannot find out URL to translate  
**not\_found:**NOT FOUND  
**number\_of\_problems\_for\_folder:**Number of problems found for this folder:  
**number\_of\_problems\_for\_host:**Number of problems found for this host  
**path\_contains\_text:**Path "[[path]]" contains the text "[[text]]"  
**provided\_by:**Provided by  
**problems\_already\_detected:**Problems already detected for this file  
**problems\_found:**problems found  
**query\_string\_contains\_text:**URL query string contains the text "[[text]]"  
**redirection\_found:**Redirection found  
**redirection\_inside\_domain:**Redirection inside domain  
**redirection\_problems:**Redirection problems  
**redirection\_to\_authorized:**REDIRECTION TO AUTHORIZED DOMAIN  
**regular\_access:**REGULAR ACCESS  
**response\_code:**Response Code  
**search\_engine:**Search Engine  
**server\_compromise:**This may be an indicator of server compromise  
**show\_data:**Show data

**summaries:**Summaries  
**summary:**Summary  
**suspicious\_content:**Suspicious content  
**test:**TEST  
**title:**Title  
**titles:**TITLES  
**translate\_using:**TRANSLATE USING [[search\_engine]]  
**unauthorized\_content:**Unauthorized content? Found: "[[hint]]". Number of occurrences: [[number]]  
**url:**URL  
**url\_index\_title:**URL and Search engine indexes and titles  
**using\_user\_agent\_of:**USING USER AGENT OF

---

---

## **Componentes Software**

---

---

## ***ESTRUCTURAS DE DATOS***

### **Search\_engine\_result**

Hash usado por los objetos de motor de búsqueda para representar un resultado

Claves:

:url	URL del resultado
:referer	URL de la búsqueda que produjo el resultado
:document_type	Tipo de documento (PDF, DOC, etc.)
:title	Título del resultado
:summary	Sumario del resultado
:cache	URL de la copia en cache del documento
:translate	URL para traducir el documento
:bot_user_agent	User Agent del robot del buscador
:s_e_object	Objeto de motor de búsqueda que produjo el resultado

### **Search\_engine\_response**

Conjunto de resultados proporcionados por un motor de búsqueda ante una consulta.

Es un array de Search\_engine\_result

### **Deferred\_engine\_result**

Hash usado por los objetos de información diferida para proporcionar información sobre URLs de cache o traducción.

Claves

:url	URL de cache o traducción
:referer	Referer a usar cuando se solicite la URL

### **BSD\_result**

Hash usado por la clase BlackSEO\_detector para representar un resultado asociado a una URL.

Las claves son los identificadores de los motores de búsqueda

Los valores son estructuras Search\_engine\_result proporcionadas por dicho buscador

## BSD\_URL\_info

Hash usado por BlackSEO\_detector para representar la información recolectada sobre una URL

Las claves son las URLs.

Los valores son los BSD\_results de esa URL.

## Report\_info

Hash para enviar datos al generador de reporte (normalmente, líneas del informe o tablas)

Claves:

:level	Valores válidos: :alert, :warning, :ok and :info
:info	Items de datos
:headers	Cabeceras para las tablas
:parameters	Parámetros par traducir los valores



# CLASES

## Clase Translator

Proporciona la traducción de mensjaes HTML .

### Métodos de instancia

- **initialize(file\_path)**  
Crea una nueva instancia. Su tabla de traducción se descargará del fichero indicado.
- **load\_table(file, add = false)**  
Abre el fichero dado por file\_path, analiza sus contenidos y los carga en la tabla de traducción. Si add es falso, la tabla se vaciará antes de cargar el fichero.
- **translate(input, tokens={})**  
Traduce una entrada (input) usando la información proporcionada por el hash "tokens", cuyo formato es:  
<token> => <value>  
Donde <token> es un Symbol y <value> una String.

### Procedimiento de traducción

Si la entrada es un Symbol, se buscará en el hash "tokens" para encontrar su traducción. Si no se encuentra, se buscará en la tabla de traducción.

Si es una String, el texto encerrado entre corchetes dobles ("[...]") se convertirá a Symbol y procesado como se indica en el párrafo anterior. will be converted to Symbol and processed as above. El resultado se usará para reemplazar los corchetes y su contenido.

### Formato de fichero de tabla de traducción

Los ficheros de tabla de traducción son ficheros de texto cuyas líneas tienen el formato:

<token>:<value>

Donde los caracteres de <token> pueden ser letras, dígitos y underscores ("\_"). Al cargarlos en la tabla de traducción, este token sera convertido a Symbol.

Se ignorarán las líneas vacías o que comiencen por "#". Puede haber espacios al inicio de la línea. Los espacios adicionales al inicio y final de la línea serán eliminados.

## Clase My\_configuration

Clase usada para proporcionar información de configuración a la aplicación

### Métodos de instancia

- **initialize(definitions)**  
Crea una nueva instancia y carga las opciones de definición (ver más abajo para más detalles)
- **read(file\_path)**  
Carga la configuración desde el fichero dado por `file_path`
- **[]=(option\_to\_assign, value)**  
Permite el uso de corchetes como si se tratara de un Hash. El índice debe ser un nombre de opción, proporcionado como Symbol
- **[](option)**  
Permite el uso de corchetes como si se tratara de un Hash. El índice debe ser un nombre de opción, proporcionado como Symbol
- **to\_s**  
Convierte la configuración en una cadena comprensible por seres humanos (...)

### Formato de definiciones

El parámetro definitions es un hash con las siguientes claves

- **:name**  
Nombre del parámetro. Sólo debe contener letras, dígitos y underscores (`_`).
- **:kind**  
Puede ser:
  - **:simple** (Un valor) - Este es el tipo por defecto
  - **:multiple** (array de valores)
  - **:pairs** (array de hashes)
- **:mandatory**  
Valor booleano o, por defecto false. Cuando es true, esta opción de configuración debe tener asignado un valor antes de usarla.
- **:type**  
**Tipo de valor**
  - **:string** Tipo por defecto
  - **:integer**
  - **:boolean.**  
Al asignar valores a una opción booleana, se considera que ésta es true si, y sólo si, el valor asignado es "y", "t", "yes" o "true". Esta comprobación no es sensible al uso de mayúsculas y minúsculas.
- **:default**  
Valor por defecto
- **:allowed**  
A veces es necesario asignar un valor a una opción de un tipo distinto al suyo.

Los valores en este array serán aceptados sin importar el tipo. Si sólo se proporciona un valor, se creará automáticamente un array a partir de él..

### **Formato del fichero de configuración**

El fichero de configuración es un fichero de texto cuyas líneas tienen uno de los siguientes formatos

- **<option>=<value>**  
Usado para opciones de tipo simple o de multiples valores. Si se trata de una opción simple, le será asignado el valor <value>. Si es de tipo múltiple, <value> será añadido a su lista de valores.
- **<option>=<sep><key><sep><value>**  
Usado para pares de valores. Las dos apariciones de <sep> deben ser el mismo carácter. Se añadirá a la lista de hashes uno del tipo{ key => value } .

Las líneas vacías o que comiencen por “#” serán ignoradas. Todos los espacios extra al inicio y fin de línea serán eliminados.

## Clase My\_HTTP\_client

Clase para realizar peticiones HTTP / HTTPS.

### Métodos de clase

- **detect\_redirection(response)**  
Compruebe si una Net::HTTPResponse conlleva redirecciones. Retorna un array con tres elementos:
  - **is\_there\_a\_redirection**  
True si la respuesta conlleva redirección
  - **redirection\_type**  
Puede ser :http or :html
  - **url\_to\_redirect**  
URL a que se redirige
- **repair\_url(url)**  
Corrige una cadena con una URL para asegurarse de que la URL es válida.

### Métodos de instancia

- **initialize(user\_agent="My HTTP Client", proxy=:ENV, proxy\_port=nil, proxy\_user=nil, proxy\_password=nil)**  
Crea un nuevo objeto con el User Agent e información de proxy proporcionados. Ruby 2 usa el proxy indicado por la variable de entorno "http\_proxy" cuando "proxy" = :ENV. Si no desea completa seguridad de no utilizar proxy, se debe asignar un valor nil
- **set\_server(protocol, server, port=:default, update\_cookies=false, usr=nil, pwd="")**  
Establece la información acerca del host al que conectar. El parámetro "protocol" puede ser :http o :https. Si "update\_cookies" es true, las cookies devueltas por el servidor serán almacenadas y utilizadas en la posterior petición. "usr" y "pwd" serán usados como credenciales para autenticación simple de HTTP.
- **change\_cookie(new\_cookie)**  
Asigna una cookie. Debe ser proporcionada como una cadena.
- **change\_user\_agent(new\_user\_agent)**  
Asigna valor al User Agent
- **change\_auth(usr, pwd)**  
Cambia las credenciales para autenticación simple de HTTP
- **post(path="/", get\_parameters={}, post\_data="", headers={}, retries=2)**  
Realiza una petición POST. Si se encuentran errores o problemas, el número máximo de reintentos vendrá dado por "retries". "get\_parameters" y "headers" son hashes de "name" => "value", siendo cadenas tanto name como value. "post\_data" puede ser un hash también, para representar un formulario, o una

cadena para representar una petición POST sin formulario

- **get(path="/", get\_parameters={}, headers = {}, retries=2)**

Hace una petición GET. Ver el método “post” para la explicación de los parámetros.

- **generic\_request(path="/", get\_parameters={}, post\_data="", headers={}, method=:get, max\_retries=2)**

Método de utilidad que permite realizar peticiones tanto GET como POST. “method” puede ser :get o :post. Ver el método “post” para la explicación de los parámetros.

## Clase My\_mailer

Clase para enviar correos

### Métodos de instancia

- **initialize(server, port=:default, cypher=:tls, user=nil, password=nil, auth\_type=nil)**

Crea una nueva instancia.

- **server**

Servidor de correo

- **port**

Puerto. Puede ser :default o un número de puerto

- **cypher**

Método de cifrado. Puede ser :ssl, :tls o :none

- **auth\_type**

Ver la documentación de Net::SMTP para detalles. Un valor típico sería :login

- **send(body, from, to, subject="", body\_content\_type="text/plain", attachments = [], cc=nil, bcc=nil, delim="\_\_\_\_MAIL\_PART\_BEGINS\_ENDS")**

Envía un mensaje

- **body**

Texto o HTML para el cuerpo del mensaje

- **body\_content\_type**

Tipo MIME para el cuerpo del mensaje. Valores típicos serían "text/html" o "text/plain"

- **attachments**

Adjuntos. Hash con estas claves:

- ◆ **:file\_name** : ruta del fichero

- ◆ **:content\_type** : Tipo MIME del fichero

- **delim**

Cadena a usar internamente para marcar las partes del mensaje.

## Clase Search\_engine

Clase para representar un motor de búsqueda genérico.

### Métodos de instancia

- **initialize(protocol, server, port=:default, path="/", query\_parameter="q", regexps = {}, paging = {}, base\_get\_parameters={}, base\_post\_parameters="", headers={}, user\_agent="Search Engine Requester", search\_engine\_bot = nil, max\_requests=5, request\_method=:get, update\_cookies=false, query\_parameter\_method=:get, proxy=:ENV, proxy\_port=nil, proxy\_user=nil, proxy\_password=nil, max\_redirs=5)**

Crea una nueva instancia.

- **protocol**  
Protocolo. Puede ser :http or :https
- **server, port y path**  
Representa la URL base para realizar las consultas
- **query\_parameter**  
Parámetro usado para enviar la consulta
- **regexp**  
Hash de expresiones regulares para encontrar las componentes de los datos de los resultados. Cada una de ellas, con la excepción de las que se incluyen en “:replaces” deben tener un subpatrón identificado con el nombre “data”
  - ◆ **:replaces**  
Hash de reemplazos <regexp> => <texto> a realizar antes de analizar la respuesta HTML del motor de búsqueda
  - ◆ **:begin\_result**  
Expresión regular para encontrar el inicio de un resultado
  - ◆ **:end\_results**  
Expresión regular para encontrar el final de todos los resultados
  - ◆ **:document\_type**  
Expresión regular para encontrar el tipo de documento del resultado
  - ◆ **:title**  
Expresión regular para encontrar el título del resultado
  - ◆ **:url**  
Expresión regular para encontrar la URL del resultado
  - ◆ **:summary**  
Expresión regular para encontrar el sumario del resultado
  - ◆ **:cache**  
Expresión regular para encontrar la URL de la cache del buscador para el

resultado

◆ **:translate**

### Expresión regular para encontrar la URL de traducción del resultado

- **paging**

Información para gestionar la paginación de resultados. Es un hash con las siguientes claves:

◆ **:parameter**

Nombre del parámetro usado para el número de página

◆ **:parameter\_method**

Forma en que este parámetro es enviado: :get, :post, :cookie

◆ **:size\_parameter**

## Parámetro para el tamaño de página

◆ :size\_parameter\_method

Cómo se envía este parámetro: :get, :post, :cookie

◆ :size\_parameter\_value

Valor a asignar al parámetro de tamaño de página

◆ **:size**

Tamaño de página especificado como un entero

- `base_get_parameters`, `base_post_parameters`, `headers`

Hashes de pares "name"=>"value" para los parámetros GET y POS

- **user\_agent**

## User Agent a usar en las peticiones

■ **search\_engine\_bot**

User Agent del robot del motor de búsqueda

- **max\_requests**

Número máximo de peticiones a realizar cuando se invoque el método "request\_all".

- request\_method

Método para enviar las peticiones: :get, :post

- **update\_cookies**

Si se debe o no actualizar las cookies con cada respuesta del servidor

- **query\_parameter\_method**

Tipo del parámetro empleado para enviar las consultas: :get, :post, :cookie

- **proxy, proxy\_port, proxy\_user, proxy\_password**

Configuración de proxy, si es necesaria. Ruby 2 usa el proxy indicado por la variable de entorno "http\_proxy" cuando "proxy" = :ENV. Si no desea completa seguridad de no utilizar proxy, se debe asignar un valor nil

- **max\_redirs**

Número máximo de redirecciones a seguir en cada petición.

- `connect_through(proxy=:ENV, proxy_port=nil, proxy_user=nil, proxy_password=nil)`

Configuración de proxy, si es necesaria. Ver detalles arriba sobre los parámetros.



- **set\_deferred(deferred)**  
Algunos motores de búsqueda no proporcionan toda la información en su página de resultados. Las URLs de cache o traducción son ejemplos de datos que pueden requerir peticiones adicionales. Para evitar hacer demasiadas peticiones, éstas deberían ser realizadas sólo si son necesarias. Un objeto de información diferida puede manejar estas situaciones. La sección “Objetos de información diferida” contiene detalles al respecto
- **deferred\_info(type, result, only\_one=false)**  
Obtiene información según es necesaria usando el objeto de información diferida. Devuelve una estructura Deferred\_engine\_result  
Parámetros:
  - **type**  
Tipo de dato solicitado: :cache, :translate
  - **result**  
Resultado del motor de búsqueda (ver definición de la estructura de datos Search\_engine\_result para más detalles)
  - **only\_one**  
True si sólo se necesita un resultado. Esto ayuda a economizar peticiones
- **request(query, page=1)**  
Hace una petición al motor de búsqueda con la consulta “query” para obtener una SERP. El número de la SERP viene dado por el parámetro “page”. Retorna una estructura Search\_engine\_response.
- **request\_all(query, pause=1, max\_results=1000)**  
Solicita todos los datos para la consulta “query” al motor de búsqueda. “pause” es el número de segundos que hay que tomar de pausa antes de realizar cada petición. “max\_results” es el número máximo de resultados necesarios. Retorna una estructura Search\_engine\_response.

## Clase Goolge\_cse

Clase para realizar peticiones a Google. Subclase de Search\_engine

### Métodos de instancia redefinidos

- **initialize(key, engine, user\_agent="G-API-RubyProg", max\_requests=3, proxy=:ENV, proxy\_port=nil, proxy\_user=nil, proxy\_password=nil)**
  - **key**  
Clave de API de Google
  - **engine**  
ID del motor de búsqueda de Google CSE
  - **user\_agent**  
User Agent a usar en las peticiones
  - **max\_requests**  
Número máximo de peticiones a realizar cuando se invoque el método "request\_all".
  - **proxy, proxy\_port, proxy\_user, proxy\_password**  
Configuración de proxy, si es necesaria. Ruby 2 usa el proxy indicado por la variable de entorno "http\_proxy" cuando "proxy" = :ENV. Si no desea completa seguridad de no utilizar proxy, se debe asignar un valor nil

## Clase Bing\_api

Clase para realizar peticiones a Bing. Subclase de Search\_engine

### Métodos de instancia redefinidos

- **initialize(key, user\_agent="B-API-RubyProg", max\_requests=3, proxy=:ENV, proxy\_port=nil, proxy\_user=nil, proxy\_password=nil)**
  - **key**  
Clave de API de Bing
  - **user\_agent**  
User Agent a usar en las peticiones
  - **max\_requests**  
Número máximo de peticiones a realizar cuando se invoque el método "request\_all".
  - **proxy, proxy\_port, proxy\_user, proxy\_password**  
Configuración de proxy, si es necesaria. Ruby 2 usa el proxy indicado por la variable de entorno "http\_proxy" cuando "proxy" = :ENV. Si no desea completa seguridad de no utilizar proxy, se debe asignar un valor nil

## Objetos de información diferida

Realmente no se trata de una clase, sino de un conjunto de métodos a implementar por los objetos usados para obtener información diferida.

### Métodos de instancia

- **initialize(user\_agent, proxy=:ENV, proxy\_port=nil, proxy\_user=nil, proxy\_password=nil) :**
  - **user\_agent**  
User Agent a usar en las peticiones
  - **proxy, proxy\_port, proxy\_user, proxy\_password**  
Configuración de proxy, si es necesaria. Ruby 2 usa el proxy indicado por la variable de entorno "http\_proxy" cuando "proxy" = :ENV. Si no desea completa seguridad de no utilizar proxy, se debe asignar un valor nil
- **connect\_through(proxy=:ENV, proxy\_port=nil, proxy\_user=nil, proxy\_password=nil)**  
Configuración de proxy, si es necesaria. Ver explicación de los parámetros más arriba
- **get\_info(type, result, only\_one=false)**  
Obtiene información bajo demanda. Retorna una estructura de datos Deferred\_engine\_result.  
Parámetros:
  - **type**  
Tipo de dato solicitado: :cache, :translate
  - **result**  
Resultado del motor de búsqueda (ver definición de la estructura de datos Search\_engine\_result para más detalles)
  - **only\_one**  
True si sólo se necesita un resultado. Esto ayuda a economizar peticiones

## Clase Report\_generator

Clase para generar informes HTML

### Métodos de instancia

- **initialize(file, overwrite=false, lang='lang/spanish.txt') :**  
Crea una nueva instancia
  - **file**  
Ruta para el fichero principal del informe
  - **overwrite**  
Si es true y el informe existe, lo sobrescribe
  - **lang**  
Ruta del fichero de idioma para las traducciones. Debe tener el formato indicado en “Formato de fichero de tabla de traducción”
- **begin(title)**  
Comienza la escritura del informe. El título del mismo vendrá dado por el contenido de “title”
- **end**  
Finaliza el informe
- **begin\_test(title)**  
Inicia una sección de TEST. Su encabezado vendrá dado por “title”.
- **end\_test**  
Finaliza una sección de TEST
- **begin\_host(host)**  
Inicia una sección de HOST. Su encabezado vendrá dado por “title”.
- **end\_host**  
Finaliza una sección de HOST
- **begin\_folder(folder)**  
Inicia una sección de DIRECTORIO. Su encabezado vendrá dado por “title”.
- **end\_folder**  
Finaliza una sección de DIRECTORIO
- **begin\_file(file)**  
Inicia una sección de FICHERO. Su encabezado contendrá el nombre del fichero indicado por “file”.
- **end\_file**  
Finaliza una sección de FICHERO
- **not\_tested(url, bsd\_result, text, params={})**  
Muestra información de una URL que no fue objeto de comprobaciones
  - **url**  
URL para la que no se realizó ninguna prueba
  - **bsd\_result**  
Estructura de datos con información sobre la URL

- **text**  
String o symbol con información sobre por qué no se hicieron pruebas para la URL. Será traducido usando un objeto de Clase Translator.
- **params**  
Parámetros para la traducción de “text”
- **problem(url, bsd\_result, analysis, to\_highlight)**  
Muestra información de una URL para la que se encontró problemas
  - **url**  
La URL
  - **bsd\_result**  
Estructura de datos con información sobre la URL
  - **analysis**  
Estructura de datos Report\_info con información sobre las pruebas realizadas
  - **to\_highlight**  
Array de cadenas a remarcar en el texto de respuestas, sumarios, etc.
- **def maybe\_ok(url, bsd\_result)**  
Muestra información de una URL para la que no se encontró ningún problema
  - **url**  
La URL
  - **bsd\_result**  
Estructura de datos con información sobre la URL

## Estructura del informe

El informe tiene las siguientes componentes

- Un fichero principal de informe HTML, con información sobre los datos proporcionados por los motores de búsqueda y su análisis, estudio de las URL y de las respuestas del servidor.
- Un fichero HTML de detalle por cada URL para la que se encontró problemas. Tiene una columna por cada test realizado. Este fichero aporta datos sobre resultados de los tests, caceceras HTTP recibidas del servidor y sus correspondientes contenidos HTML.

El informe usa JavaScript para permitir mostrar u ocultar elementos y para el ajuste del ancho de las columnas.

Los ficheros de detalle se almacenan en un subdirectorio ubicado en el mismo directorio que el fichero principal del informe.

El nombre del directorio para los ficheros de detalle es el mismo del fichero principal del informe seguido de una terminación “.dir”.

## Clase BlackSEO\_detector

Clase que analiza modificaciones no autorizadas de webs usadas para técnicas de BlackSEO

### Métodos de instancia

- **initialize(search\_engines, user\_agent="BlackSEO-detector", proxy=:ENV, proxy\_port=nil, proxy\_user=nil, proxy\_password=nil, verbose=true)**

Crea una nueva instancia

- **search\_engine**

Hash de "name"=>search\_engine\_object

- **user\_agent**

User Agent a utilizar en las peticiones

- **proxy, proxy\_port, proxy\_user y proxy\_password**

Configuración de proxy, si es necesaria. Ruby 2 usa el proxy indicado por la variable de entorno "http\_proxy" cuando "proxy" = :ENV. Si no desea completa seguridad de no utilizar proxy, se debe asignar un valor nil

- **investigate(queries, text, report\_generator, report\_name='Test', hard\_check=false, max\_redirects=5, valid\_domains=[], ignore\_files=[], max\_report=20, max\_host=12, max\_folder=5)**

Realiza una prueba

- **queries**

Hash de "search engine name" => "query" con las consultas a hacer a cada motor de búsqueda

- **text**

Array de textos usados como indicio de modificaciones no autorizadas

- **report\_generator**

Objeto a usar para generar el informe. Ver "Clase Report\_generator" para más detalles

- **report\_name**

Título principal del informe

- **hard\_check**

Si es falso, los tests para una URL finalizarán tan pronto como un test revele la existencia de problemas. En otro caso, todos los tests serán realizados para todas las URLs

- **max\_redirects**

Número máximo de redirecciones a seguir cuando se realizan peticiones HTTP

- **valid\_domains**

Array de dominios a incluir en el reporte.

Las URLs de dominios no incluidos en esta lista serán ignoradas.

Las redirecciones a dominios no incluidos en esta lista producirán una alerta en el informe.

- **ignore\_files**  
Array de expresiones regulares. Las URLs que sigan cualquiera de estos patrones no serán objeto de comprobaciones
- **max\_report, max\_host, max\_folder**  
Número máximo de URLs a comprobar por informe, host y directorio, respectivamente.

# Índice

Guía del usuario.....	1
¿Qué es BlackSEO Detector?.....	2
¿Qué hace BlackSEO Detector?.....	2
Características.....	2
Uso.....	3
Fichero de Configuración.....	3
Ejemplo de fichero de configuración.....	9
Añadir funcionalidades.....	11
Añadir soporte para motores de búsqueda.....	12
Añadir soporte para idiomas.....	12
Componentes Software.....	15
ESTRUCTURAS DE DATOS.....	16
Search_engine_result.....	16
Search_engine_response.....	16
Deferred_engine_result.....	16
BSD_result.....	16
BSD_URL_info.....	17
Report_info.....	17
CLASES.....	18
Clase Translator.....	18
Métodos de instancia.....	18
Procedimiento de traducción.....	18
Formato de fichero de tabla de traducción.....	18
Clase My_configuration.....	19
Métodos de instancia.....	19
Formato de definiciones.....	19
Formato del fichero de configuración.....	20
Clase My_HTTP_client.....	21
Métodos de clase.....	21
Métodos de instancia.....	21
Clase My_mailer.....	23
Métodos de instancia.....	23
Clase Search_engine.....	24
Métodos de instancia.....	24
Clase Goolge_cse.....	27
Métodos de instancia redefinidos.....	27
Clase Bing_api.....	27
Métodos de instancia redefinidos.....	27
Objetos de información diferida.....	28
Métodos de instancia.....	28
Clase Report_generator.....	29



Métodos de instancia.....	29
Estructura del informe.....	30
Clase BlackSEO_detector.....	31
Métodos de instancia.....	31
Índice.....	33