



Ekioh Configuration

Copyright 2008-2016 Ekioh Ltd.

Release 3.6.0

07 January 2016

Contents

1	Introduction	2
2	Configuration File	3
2.1	Setting Configuration Options	3
2.2	Configuration File Format	3
3	Configuration Options	4
3.1	Generic Configuration Options	4
3.2	SVG Configuration Options	10
3.3	Browser Configuration Options	13
4	Command Line Operation	17
4.1	Command Line Options	17
5	Environment Variables	18
5.1	Environment Variables	18
5.2	ICU Data Location	18
6	SSL Client Certificates	19
6.1	Configuration	19
6.2	Certificate format	19

Introduction

The Ekioh UI Engine and TV Browser are highly configurable and can be tailored to specific needs using a configuration file and command line arguments.

In addition, environment variables can be used for proxy settings, plugin and font directory locations.

This document describes how to configure the Ekioh UI Engine and TV Browser, set the environment variables and accept client certificates.

Further documentation can be found on the Ekioh Releases Page at <https://support.ekioh.com/releases/>.

Configuration File

The Ekioh UI Engine and TV Browser use default configuration settings that can each be overridden by using the Ekioh Configuration File, ekioh.cfg. This in turn can be overridden by additional configuration files and command line arguments. This section shows where the configuration file is located, how to override it and the default configuration settings.

2.1 Setting Configuration Options

The default configuration settings can be overridden by using entries in a configuration file, ekioh.cfg. The default location for ekioh.cfg is the current directory. (This is not necessarily the same directory as the ekioh binary).

Additional configuration files can also be used to make additions or amendments to the default and configuration file settings. To specify an additional configuration file, type:

```
ekioh --config-file <config file>
```

All configurations in this additional file are added to those in the default configuration file and if a parameter is set more than once, the one in the additional file is used.

Some configuration options can be set via the command line as described in the next section. Command line settings are added to the defaults and those set in the configuration file in the default location. If the same parameter is given on the command line and in a user specified configuration file, the last to appear on the command line is used.

2.2 Configuration File Format

Configuration file entries are of the format:

```
#comment  
<name>:<value>
```

Configuration Options

Some configuration items can be changed dynamically at runtime via the porting layer.

3.1 Generic Configuration Options

Configuration Option	Default	Description
application.background.argb	0xffffffff	Background colour (in ARGB format) for the application to plot on top of. The alpha component is plotted to the screen, rather than blended, allowing video to show through behind the document contents if it is not 0xff. A value of none ensures no background is plotted. This is only of use if some other application controls the framebuffer, otherwise the screen will look corrupt. (2.2.1.4 onwards).
application.default	[blank]	Specifies the default application (SVG is default) (3.4.1 onwards). <ul style="list-style-type: none"> browser: Forces Twin Engine browser builds into TV Browser mode.
application.geometry	720x576	Specifies the size of the window in pixels for desktop builds. For example, “1280x720” for 720p HD. The default window size is SD PAL. (2.0.1 onwards).
application.homepage	[blank]	Homepage URL to load at startup. (2.0.1 onwards).
application.language.default	en	Specifies the default language, as defined in rfc4646. (2.2.7.2 onwards).
Continued on next page		

Table 3.1 – continued from previous page

Configuration Option	Default	Description
application.verbose	false	Enables verbose output to the console, useful for tracking down memory or file I/O issues. (2.1.8.2 onwards).
application.websocket	true	Enable WebSocket support for the application. (2.2.7.7 onwards).
font.accelerate	true	Hardware-accelerate text (FreeType builds only). Can be disabled on some platforms where the port restricts the performance. Puts an extra burden on the font cache. (2.2.7 onwards).
font.cache.capacity	2MB	Size of the font cache (FreeType builds only). Units are bytes (no suffix), KB or MB. Prior to version 3.5 this defaulted to 1MB. (2.0.1 onwards).
font.default	[blank]	The name of the default (fallback) font. This is used if the font name is not found or not specified. If this option is not specified, the first font loaded will be the default. This is the first font included in the build or, if none, the first found in the search path. (2.1.1 onwards).
font.fallback	[blank]	A comma-separated list of fallback fonts (as per the CSS property 'font-family'). The fallback font is used when the specified or default font is missing a required glyph. (2.2.5.7 onwards).
font.hinting	false	Configure font hinting. (2.2.2.10 onwards for FreeType, 3.5.3 onwards for iType).
font.path	[blank]	Directory path to search for dynamically loadable fonts. Multiple paths can be specified using the platform's path separator character (':' on Unix systems). (2.1.1 onwards).
geolocation.preset	[blank]	For platforms with no geolocation support, this option specifies the latitude, longitude (degrees) and accuracy (metres) as space-separated numbers. e.g. for a 5km radius from central Cambridge, '52.205282 0.117811 5000'. (2.1.8.4 onwards).
graphics.filters.accelerate	true	Enables the use of hardware accelerated filter effects (if supported). When false, all filter effects are performed in software. (2.2.6.3 onwards).
Continued on next page		

Table 3.1 – continued from previous page

Configuration Option	Default	Description
network.cookie.database	[blank]	The filename of any cookie database. If present, this is loaded on EKLibrary startup, and saved on shutdown. The file may be read-only, in which case it is not overwritten on shutdown. No error is generated if the file does not exist, or cannot be read. (2.0.1 onwards).
network.digest.database	[blank]	The filename of any digest database for use by digest authentication. If present, this is loaded on EKLibrary startup, and saved on shutdown. The file is expected to be populated via JavaScript APIs rather than by hand. The password is stored as an MD5 digest rather than plaintext. (2.0.1 onwards).
network.http.accept.language	[blank]	The given string is used as the 'Accept-Language' option in HTTP and HTTPS requests. (2.2.4.3 onwards).
network.http.gzip	true	Enable notification of gzip support in HTTP requests. With this false, gzip responses are never requested but are still accepted and uncompressed if received. (2.0.3 onwards).
network.http.keepalive	true	Support HTTP keep-alive. When disabled, connections are closed when their request completes. (2.1.8.5 onwards).
network.http.keepalive.timeout	0	HTTP keep-alive timeout, in milliseconds. Connections which have been idle (no requests) for this period of time will be closed. A timeout value of 0 will cause connections to be kept alive indefinitely. (3.1.2 onwards).
network.http.max-connections	30	The total number of concurrent asynchronous connections (all servers). (2.2.4.1 onwards).
network.http.max-connections-per-server	15	The total number of concurrent asynchronous connections per server. (2.2.4.1 onwards).
network.http.proxy	[blank]	A URL to a proxy used for http traffic e.g. http://192.168.0.1:3128 . If this option is blank the platform's proxy settings are used, e.g. <code>http_proxy</code> environment variable when running on Linux. (2.2.5.8 onwards).
Continued on next page		

Table 3.1 – continued from previous page

Configuration Option	Default	Description
network.http.noproxy	[blank]	A comma-separated list of sites, domains and addresses to not proxy. To match all hosts on a given domain, prefix the domain with a dot (e.g. '.ekioh.com' to match both www.ekioh.com and support.ekioh.com). To match a range of IP addresses use CIDR format (e.g. 192.168.0.0/16 or 2001:db8::/48). (3.0.1 onwards).
network.http.redirection-limit	20	The total number of HTTP redirects before aborting the connection. (2.1.3 onwards).
network.https.proxy	[blank]	A URL to a proxy used for https traffic e.g. https://192.168.0.1:3128. If this option is blank the platform's proxy settings are used, e.g. https_proxy environment variable when running on Linux. (2.2.5.8 onwards).
network.ipv6	false	IPV4 sockets are created by default, if enabled IPV6 sockets are created. (2.2.2.5 onwards).
plugin.mime-typesautoload	[blank]	A comma separated list of mime-types specifying the plugins which are to be automatically instantiated on every page. Whitespace is not allowed within the list of mime-types. (2.2.5.9 onwards).
plugin.mime-types.force-solid	[blank]	A list of plugin MIME types and colours in the form < MIME type >,< hex colour >,< MIME type >,< hex colour > for which the browser should draw a solid rectangle instead of passing paint events to the plugin. For example video/broadcast,0x00000000. (3.4.3 onwards).
plugin.path	[blank]	Directory path to search for dynamically loaded plugin libraries. Multiple paths can be specified using the platform's path separator character (': on Unix systems). If plugin.path is blank, the EK_PLUGIN_DIR environment variable is used instead. If EK_PLUGIN_DIR is not specified then no plugins are loaded. (2.2.2.4 onwards).
ssl.cafile	[blank]	A file of CA certificates in PEM format. See OpenSSL for more information. (2.0.1 onwards).

Continued on next page

Table 3.1 – continued from previous page

Configuration Option	Default	Description
ssl.capath	[blank]	A directory containing CA certificates in PEM or DER (.der extension) format (one certificate per file). The OpenSSL <code>c_rehash</code> utility must be used to generate appropriately named links to the certificate files. See OpenSSL for more information. (2.0.1 onwards).
ssl.certs.default.enable	true	Enable a set of built-in certificates from common root authorities. (2.2.2.12 onwards).
ssl.certs.js.install	false	When true, allows scripts to install server certificates locally (for that session). This enables a fully trusted security path to external servers without requiring the certificates to be installed on the STB in its flash image. Note that incorrect setup can make this a security risk; all documents must be loaded over HTTPS with verified certificates. (2.2.2.12 onwards).
ssl.client.certs.path	[blank]	A directory containing customised .pem files which each contain a client certificate and associated private key and password. (2.2.9.1 onwards).
ssl.compression.enable	false	Enabling compression will speed up connections but also make them less secure. (2.3.1 onwards).
ssl.crlfile	[blank]	The location of any CRL file to use for certificates specified by both <code>oipf.widget.cafile</code> and <code>ssl.cafile</code> or <code>ssl.capath</code> . (2.1.0 onwards).
ssl.disable.tls	false	When enabled, the application will request SSLv3 connections instead of TLS. (2.2.8.7 onwards).
ssl.rc4.enable	false	When enabled, will allow the use of less secure RC4 ciphers. https://tools.ietf.org/html/rfc7465 (3.3.3 onwards).
ssl.ssl3.enable	false	When enabled, will allow SSLv3 connections to be used. (3.1.1 onwards).
ssl.validate.hostname	true	Validate the hostname of the remote server against the Subject Alternative Name or Common Name fields of the server certificate. (2.2.4.7 onwards).

Continued on next page

Table 3.1 – continued from previous page

Configuration Option	Default	Description
ssl.verify	true	Verify the server certificate against the internal certificates. It is strongly recommended not to set this to false in deployment situations, but it can be useful during development. A value of false is equivalent to automatically clicking 'OK' on a web browser's dialogue querying an untrusted certificate. (2.0.1 onwards).
svg.database.cache.size	2MB	The size of the memory cache that should be used for handling database data. This affects both WebStorage and WebSQL. Units are bytes (no suffix), 'KB' or 'MB'. (2.2.7 onwards - versions prior to 3.1.1 only accept this value in bytes without a suffix).
svg.database.maxquota	5MB	The maximum quota allowed for each database. Units are bytes (no suffix), 'KB' or 'MB'. (2.1.8.2 onwards - versions prior to 3.1.1 only accept this value in bytes without a suffix)
svg.database.path	[blank]	The directory path where database files used by the UI Engine should be stored. This includes WebSQL databases. NB: This does not include WebStorage, which has its own configuration option. (2.1.8 onwards).
svg.database.trusteddomains	[blank]	A comma-separated list of trusted domains which are allowed to manage the database origins. Note that the 'file://' domain is referred to as 'localhost' in this list. (2.1.8.2 onwards).
svg.webstorage.database	[blank]	The directory in which to store WebStorage databases. If the directory does not exist, it will be created on first use. If blank, the 'window.localStorage' object will not be created. This behaviour alters if the 'svg.webstorage.legacy.enable config' option is enabled (see the description of that option for more details). (2.2.8.6 onwards).
svg.webstorage.legacy.enable	false	Allows selection of the legacy (pre 2.2.5.7) WebStorage database format. Note that this option is ignored if a newer database format is found. When enabled, the 'svg.webstorage.database' config option specifies the name of the database file and not a directory. (2.2.6.3 onwards).
Continued on next page		

Table 3.1 – continued from previous page

Configuration Option	Default	Description
svg.webstorage.originator.events	false	WebStorage StorageEvents are no longer sent to the originator of WebStorage changes as described in the W3C spec. Setting this option to ‘true’ will reinstate the legacy behaviour of sending StorageEvents to the originator of changes. (2.2.9 onwards).
win32.font.enable	false	Enables the Windows font manager in builds that include it. When false, the default built in font manager is used. (2.2.7 onwards).

3.2 SVG Configuration Options

SVG Configuration Option	Default	Description
application.threads.cores	0	Selects number of cores available for task threads. “0” means run everything on the main thread (no threads), and higher numbers run tasks in parallel to the main thread. A value of “1” can be used on single-core CPUs to decode images in a separate thread to redraw, possibly improving animations. (3.5.0 onwards).
oipf.nvstorage.basepath	[blank]	Basepath for the persistent NV storage that will be prefixed to all filepaths sent to the NV file methods. There should be no trailing slash (‘/’) on the basepath. (2.0.2 onwards).
oipf.ramstorage.basepath	[blank]	Basepath for the RAM/local FS storage that will be prefixed to all filepaths sent to the RAM/local FS file methods. There should be no trailing slash (‘/’) on the basepath. (2.0.2 onwards).
oipf.widget.cadigitalsig	false	States whether CAs and Sub-CA certificates must have the DigitalSignature key usage bit to be verified as valid. (2.0.2 onwards).
oipf.widget.cafile	[blank]	The location of the top level CA Authority certificate used to verify the certificate chain passed to the OIPF Application Manager when a Widget install occurs. (2.0.2 onwards).
Continued on next page		

Table 3.2 – continued from previous page

SVG Configuration Option	Default	Description
oipf.widget.maxsize	15MB	Maximum installable Widget size. This is tested as the size of the .wgt file to install, and not the decompressed Widget. Units are bytes (no suffix), 'KB' or 'MB'. (2.0.2 onwards - versions <i>prior to 2.1.1</i> only accept this value in MB without a suffix).
svg.errorpage	[blank]	URL of error page to display on UI Engine fetch failure. Details of the error, if available, are supplied as semicolon separated key-value pairs in the query field of the URL: <ul style="list-style-type: none"> • httpStatusCode=<number> • httpStatusText=<string> E.g. file:///etc/error.svg? httpStatusCode=404; httpStatusText=NotFound (2.1.8 onwards).
svg.font.size.default	16	Default font size for regular text, in pixels. (2.2.4.7 onwards).
svg.framerate	50	Number of frames per second to attempt to frame limit to. (2.2.4.7 onwards).
svg.gc.periodic	5000	Default timeout, in milliseconds, between periodic checks for whether a garbage collect should be attempted. This does not mean a garbage collection occurs this frequently. (2.1.8.4 onwards).
svg.image.accelerate.bmp	false	Accelerate all BMP images, regardless of buffered-rendering property. (2.1.7 onwards).
svg.image.accelerate.gif	false	Accelerate all GIF images, regardless of buffered-rendering property. (2.1.7 onwards).
svg.image.accelerate.jpeg	false	Accelerate all JPEG images, regardless of buffered-rendering property. (2.1.7 onwards).
svg.image.accelerate.png	false	Accelerate all PNG images, regardless of buffered-rendering property. (2.1.7 onwards).
svg.image.load.timeout	0	Timeout value used when fetching images, given in milliseconds. If an image hasn't been loaded in this time it is considered to have failed and aborted. A timeout value of 0 means no timeout is set. (2.2.5.9 onwards).
Continued on next page		

Table 3.2 – continued from previous page

SVG Configuration Option	Default	Description
svg.jsheap.full.warning	64KB	Amount of free memory available in the JavaScript heap before a low memory event is sent to the document (after a garbage collect). Units are bytes (no suffix), KB or MB. (2.1.7 onwards - <i>versions prior to 3.1.1 could match KB and MB case insensitively</i>).
svg.jsheap.size	4MB	Size of the UI Engine's JavaScript heap. Units are bytes (no suffix), KB or MB. The value is automatically doubled on 64bit systems. (2.0.1 onwards - <i>versions prior to 3.1.1 will only accept this value in bytes and MB</i>).
svg.layers.debug	false	Enables debug borders and paint counters for SVG layers. (2.2.7 onwards).
svg.network.connection.security	true	Set the security for the legacy SVG Connection object. When enabled, a connection can only be made to the server at the document's origin. When disabled, a connection can be made to any server, regardless of the document's origin. (2.1.8.4 onwards).
svg.network.useragent		Allows the UI Engine user agent string to be overridden. This option may be changed at runtime by the porting layer. The following substitutions are supported: <ul style="list-style-type: none"> • %b Ekioh build number • %d Build date • %e Ekioh version number • %% A literal % character (2.2.4.10 onwards).
svg.profiling	false	Output profiling information for various operations (see Ekioh UI Engine Embedded Tools Document for more details). (2.1.8 onwards).
svg.webdriver.enabled	false	Enable Selenium WebDriver support in the UI Engine. (2.2.6 onwards).
svg.webdriver.highlight.colour	0x8033b5e5	Colour (in ARGB format) of the SVG DOM Inspector highlight. (2.2.8.7 onwards).
svg.webdriver.highlight.time	-1	Time in milliseconds for which the SVG DOM Inspector highlight should be shown. Set to -1 to display constantly, or 0 to hide it completely. (2.2.8.7 onwards).
Continued on next page		

Table 3.2 – continued from previous page

SVG Configuration Option	Default	Description
svg.webdriver.log.size	100	Maximum number of entries which can be kept in the Selenium WebDriver log buffer. If this size is exceeded then new log entries will overwrite the old ones. (3.1.2 onwards).
svg.webdriver.poll.interval	100	Time interval in milliseconds between element checks when waiting for a search timeout. (2.2.7 onwards).
svg.webdriver.port	4444	Port on which the Selenium WebDriver should listen for incoming connections. (2.2.6 onwards).

3.3 Browser Configuration Options

The following configuration options are available for browser enabled builds:

Browser Configuration Option	Default	Description
browser.brokenimage.enabled	true	Display the broken image icon when an image cannot be loaded. (2.2.4.4 onwards).
browser.cache.capacity	4MB	Size of the browser's cache. Units are bytes (no suffix), 'KB' or 'MB'. (2.0.1 onwards - versions prior to 3.1.1 only accept this value in MB or bytes).
browser.caret.width	1	Width in pixels of the browser text caret. (2.2.8.2 onwards).
browser.compositing.enabled	true	Enables or disables accelerated compositing in the browser. (2.2.7.7 onwards).
browser.compositing.3d.disable	false	Force the use of the 2D compositing in browser builds that support 3D compositing. (2.2.6 onwards).
browser.compositing.debug	false	Enable display of debug borders and repaint counters for composited content in browser builds that support compositing. (2.2.5.9 onwards).
browser.compositing.force	false	Force compositing in builds that support it, even if the content doesn't require it. (2.2.6 onwards).
Continued on next page		

Table 3.3 – continued from previous page

Browser Configuration Option	Default	Description
browser:errorpage	[blank]	URL of local error page to display on browser load failure. Details of the error, if available, are supplied as semicolon separated key-value pairs in the query field of the URL: <ul style="list-style-type: none"> • httpStatusCode=<number> • httpStatusText=<string> E.g. file:///etc/error.html? httpStatusCode=404; httpStatusText=NotFound (2.2.8.5 onwards).
browser:font.size.default	16	Default font size used by the browser to display regular text. (2.2.1.3 onwards).
browser:font.size.fixed.default	16	Default font sized used by the browser to display monospaced text. (2.2.1.3 onwards).
browser:font.size.logical.minimum	5	Minimum logical font size used by the browser to display text. Applies when the browser view is zoomed out. (2.2.1.3 onwards).
browser:font.size.minimum	5	Minimum font size used by the browser to display text. (2.2.1.3 onwards).
browser:image.accelerate.all	false	Force all bitmap images to be accelerated, with system memory fallback if out of graphics memory. (2.2.4.7 onwards).
browser:image.auto-load	true	Allows auto-loading of images to be disabled. This option may be changed at runtime by the porting layer. Any images already loaded and cached are still displayed, but new images are not fetched or displayed. Reenabling the option refetches any uncached images and redisplays all images. (3.2.1 onwards).
browser:image.max-pixels	[blank]	Limit the maximum number of pixels per decoded GIF, JPEG & PNG image in the browser. Units are pixels, with an optional K or M suffix. There is no limit if this option is not specified. (3.1.1 onwards).
browser:image.rendering.quality	medium	Bitmap image scaling quality in the browser. <ul style="list-style-type: none"> • low: nearest-neighbour scaling • medium: bilinear interpolation • high: currently equivalent to medium (2.2.5.8 onwards).

Continued on next page

Table 3.3 – continued from previous page

Browser Configuration Option	Default	Description
browser:image.rendering.progressive	true	Enables or disables progressive rendering of images. (2.2.7.7 onwards).
browser:media.controls.keys	false	Enable HTML5 video support for media control keys. (2.2.7.4 onwards).
browser:mediasource.enabled	false	Enable HTML5 media source extensions. This may alter the page behaviour if not fully implemented in the porting layer. (2.3.1 onwards).
browser:mediasource.buffer.maxsize	16MB	Maximum size for a media source buffer. Units are bytes (no suffix), KB or MB. Supported in WebKit600 or later. (3.4.2 onwards).
browser:mediasource.buffer.video.percentage	95	Percentage size of a media source buffer which is assumed to be for video. Value must be in range 0-100. Supported in WebKit600 or later. (3.5.3 onwards).
browser:navigation.enabled	true	Enables browser navigation by mapping DOM_VK_BACK and DOM_VK_FORWARD. (2.3.4 onwards).
browser:network.useragent		Allows the HTML browser user agent string to be overridden. This option may be changed at runtime by the porting layer. The following substitutions are supported: <ul style="list-style-type: none"> • %e Ekioh version number • %p Platform name • %w WebKit version number • %% A literal % character (2.2.2.9 onwards).
browser:offlineappcache.path	[blank]	Location to store resources for HTML Offline Web applications. The feature is disabled if this is left blank. (2.2.6 onwards).
browser:offlineappcache.quota	[blank]	Quota for HTML Offline Web applications in bytes. There is no quota if this is left blank. (2.2.6 onwards).
browser:outline.force	false	Display focus outline in browser, even on sites that disable it with CSS outline:none. (2.2.2.4 onwards).
browser:plugins.disable	false	Disable loading of plugins in the browser. (2.2.2.9 onwards).
Continued on next page		

Table 3.3 – continued from previous page

Browser Configuration Option	Default	Description
browser:scripting.bridge.enabled	false	Determines whether the SVG and browser JavaScript runtimes can communicate. (3.3.4 onwards).
browser:spatialnavigation	true	Determines whether WebKit's spatial navigation mode is activated. If enabled, it is possible to select focusable page items using the up/down/left/right keys. (3.1.2 onwards)
browser:stylesheet.user	[blank]	URL of user style sheet for the browser. (2.2.4.10 onwards).
browser:text.selection.enabled	true	Enables or disables text selection in the browser. This option may be changed at runtime by the porting layer. While enabled, touch scrolling will be disabled. (2.2.4.2 onwards).
browser:video.aspectratio.preserve	true	Enables or disables the preserving of the aspect ratio for a video source when embedded into a <video> element. This is only supported in versions of WebKit up to and including WebKit537; use the CSS object-fit property to control the video aspect ratio from WebKit600. (2.2.3.8 onwards).
browser:viewport.configurable	false	Enables the viewport meta tag feature. This allows the content to change the scale and dimension of the user-agent's viewport. (2.2.2.3 onwards).
browser:viewport.device.width	[blank]	Sets the device width used for calculating the viewport size and scale when the viewport is configurable. If this option is not set then the current size of main frame is used for the calculation. (2.2.2.3 onwards).
browser:webgl	false	Enables WebGL in the browser. (2.2.7 onwards).
browser:webgl.multisampling	false	Enables multisampling (anti-aliasing) for WebGL. This is currently always disabled for OpenGL ES 2.0 platforms. (2.2.7 onwards).
font.downloadable	true	Downloadable font support (currently browser only). (2.2.1.10 onwards).

Command Line Operation

Some configuration items have been mapped onto command line options which can be run with arguments as described below.

Command line options must begin with either a single or double dash.

If more than one command line option is specified for the same parameter, the last one to appear on the command line is used with the exception that `-config-file` and `-optional-config-file` may be specified multiple times.

If more than one configuration file is specified on the command line, the configuration options from all files are used with conflicts handled as described in the Setting Configuration Options section.

Several Ekioh windows can be opened by listing them as arguments. An argument that does not begin with either “-” or “- -” will signify the end of the command line options and all following arguments will be considered to be URLs or documents to be loaded.

4.1 Command Line Options

Syntax	Action
<code>-config-file <config file></code>	Specify additional config file and quit if not found.
<code>-optional-config-file <config file></code>	Specify additional config file and continue if not found.
<code>-geometry <width>x<height></code>	Set application geometry (application.geometry).
<code>-url <URL></code>	Set URL to load at start up (application.homepage).
<code>-verbose</code>	Run in verbose mode (application.verbose).
<code>-version</code>	Display Ekioh version.
Browser enabled builds	
<code>-browser</code>	Forces Twin Engine browser builds into TV Browser mode. Equivalent of configuration option ‘application.default:browser’
<code>-browser-cache-capacity <size></code>	Set browser cache capacity (browser.cache.capacity).

Environment Variables

The following environment variables are supported.

5.1 Environment Variables

The environment variables below can be used to control the behaviour. They each have equivalent configuration options which can be set in the configuration file. If the equivalent is specified in the configuration file, it will overwrite the environment variable.

Environment Variable	Configuration Equivalent	Operating System
http_proxy	network.http.proxy	Linux, Mac OS X
https_proxy	network.https.proxy	Linux, Mac OS X
no_proxy	network.http.noproxy	Linux, Mac OS X
EK_PLUGIN_DIR	plugin.path	Linux, Windows, Mac OS X

5.2 ICU Data Location

Browser enabled builds use a built in ICU library for Unicode text handling which supports US-ASCII, ISO-8859-1 and Unicode. To use a different ICU Data file, the environment variable `ICU_DATA` must be set to show its location.

```
export ICU_Data =<ICU Data Directory>
```

SSL Client Certificates

It is possible to configure Ekioh so that it accepts client certificates. When a secure server requests one during an SSL handshake, those specified will be analysed and if any match, will be sent to the server.

6.1 Configuration

The `ssl.client.certs.path` configuration option specifies the directory in which the client certificates can be found. These files should have the `.pem` extension and each contain only one certificate. Any certificates that have expired will be ignored.

6.2 Certificate format

The format of the certificate files is customised slightly so that all necessary information is stored conveniently in one place. The following three pieces of information are expected to exist within each file:

Certificate

This will be a Base64 encoded set of data enclosed by the -----BEGIN CERTIFICATE----- and -----END CERTIFICATE----- statements. It is also a requirement that the certificate is signed with an organisation name as this is used for matching against server certificates. This organisation name is specified in the `subject` and `issuer` lines of the certificate and must therefore also exist within the file.

Private Key

This will be a Base64 encoded set of data enclosed by the -----BEGIN RSA PRIVATE KEY----- and -----END RSA PRIVATE KEY----- statements. The key encryption will also be detailed within these statements.

Password

This is necessary to decode the certificate and private key and must be specified in the format: “`password:<password>`”.

These three components can be specified in any order within a `.pem` file.