
swGateKeeper™ v 1.0.1*For DotNetNuke & Smith Cart*

Thank you for purchasing (or downloading) swGateKeeper. swGateKeeper was designed to provide web administrators with a way to control the distribution of downloadable content. There are three “top level” features that can assist you.

- Controlling the number of downloads per order detail item
- Controlling the amount of time during which a customer can download an item (from the date of their first attempt)
- Stamping PDF files with customer specific information so that you could track the files after they are released into the wild – and also provide a disincentive to having customers pass them on.

swGateKeeper is not a DNN module, but is integrated with Smith Cart to read data from the tables; order, product and customer tables. Part of the reason for this is that requests will not normally come from logged in users, and requiring a login would put additional burden on customers.

swGateKeeper uses the open source iTextSharp component to provide PDF stamping features. While this component can be somewhat challenging to work with, we’ve done the heavy lifting in order to provide this valuable feature without requiring you to purchase a 3rd party pdf component.

swGateKeeper is useful for managing the delivery of ZIP, EXE, etc files in addition to PDF files. Any file extension not defined for PDF stamping will simply be transmitted subject to access & time restrictions.

Installation

The swGateKeeper.exe file is a self extracting zip file. It should be extracted into the DNN webroot. Files will be placed into the /bin folder and also into a new folder called /swGateKeeper.

swGateKeeper has a 5 day trial period. This should be enough time to determine whether the product will suit your needs. During the trial period the download count is limited to 1 per user/item, and the count cannot be reset. In addition, all pdf files will contain a conspicuous additional stamp. When the trial period has expired, all functionality will be locked. You can purchase a license for a domain through Smith-Consulting.com and will be provided with a license file to be placed in the swGateKeeper folder. This will unlock the product for the specified domain.

Quick Start

To test that swGateKeeper installed correctly, open up a browser window, and navigate to:

<http://www.DnnDomain.com/swGateKeeper/Download.aspx?g=1>

- Substitute your domain for DnnDomain
- The result should be a sample.pdf file with predefined stamps.

You can get setting information by appending “&debug=code” (where *code* is a string you supply) to the end of the url. However, you must first specify a debug code in the “my-settings.aspx” file (see “Configuring

swGateKeeper” for details). We decided to not include a default code as it would be too tempting to simply leave it unchanged.

How to setup your Smith Cart to work with swGateKeeper

To use swGateKeeper to deliver digital content from the Smith Cart, you must setup the following fields in the “Downloadable Product Setup” section of the Smith Product Setup page.

Field	Usage	Example
Download path	The relative url path to the swGateKeeper download aspx page, plus “?g=” which represents the parameter that will contain the GUID (see below) for the detail record.	/swGateKeeper/Download.aspx?g=
Product path	The relative url path to the file to be delivered. If no path is specified, swGateKeeper will assume that it is in the “default” folder as specified in the configuration settings.	/portals/0/Files/sample.pdf
Auto Generate GUID	You must check this box in order for the Smith Cart to generate a globally unique id for the order detail record. This GUID acts as both a key to the record and a password (since it is a 64 bit #).	[X]

That’s basically it. When the Smith Cart creates the order and order detail record, emails and confirmation page, the download link will go to swGateKeeper, which will manage the delivery of the content.

What Happens when swGateKeeper is used

When your users click on your download url (to swGateKeeper/download.aspx), swGateKeeper looks up the order using the guid supplied by the Smith Cart. We use a guid for security, as simply using the detail id would make it easy for outsiders to attempt to download items using similar detail ids (which are sequential). Adding a password would work, but the use of a guid essentially provides password protection – along with the record – in one field.

If the detail item is found, we check our history of download activity for the item. If it has been not reached the defined maximum/timeframe, we collect information from the shopping cart about the item and the purchaser and return it to the application. The next step is to lookup the file extension of the product against predefined PDF stamping templates. If none are found, we treat the file as a simple file transfer request (with the location of the file hidden behind the Download url). If the file extension if for a known PDF template, the template, data and source PDF file are sent to the stamper, where the data is stamped onto each page. Our database is updated with the request and then the resulting file (pdf or otherwise) is delivered to the user.

Configuring swGateKeeper

There are two configuration files: swGateKeeper-Defaults.aspx, and _my-settings.aspx. These files have .aspx extensions so that their contents cannot be viewed in a browser. If you rename _my-settings.aspx to my-settings.aspx (ie., remove the leading “_”), it will be used to override the default settings, and will not be

overwritten should you install an update in the future. Any setting in the swGateKeeper-defaults.aspx file can be overridden by including a corresponding entry in my-settings.aspx.

The format of the settings files is a bit different from what most people are use to. It is neither XML or INI. Instead, it uses the concept of templates that contain fields and data. Templates begin with “[*templateName]” and end with “[/*]”. Fields begin with “[FIELD:” and end with “[/]”.

There are two sections: [*main] and [*info.page]. [*main] contains the configuration settings, and [*info.page] contains the html used to provide error and/or debug feedback.

Specific stamp page definitions are defined in a field, and parsed when loaded. The easiest way to create your own templates is to copy the sample template included with the product and modify it (by placing it into my-settings.aspx).

The following table contains all of the fields that can be set.

General		
	max_downloads_per_order	After the user has downloaded the file this number of times, they will be shown a message that the download limit has been exceeded. Note: you can reset the download count to allow them access by using the reset key (see below). Unregistered copies of swGateKeeper are limited to 1 download per file, with no resetting.
	max_days_to_download_order	Once the customer downloads the file the first tme, they can continue downloading, up to the maximum number of times (above), for the number of days specified here. This can be useful in the case where you want to allow many download attempts for a relatively short period of time.
	downloads_folder	The folder where swGateKeeper will look for source files that do not have a relative url (no “/”). Defaults to the swGateKeeper folder.
	debug_key	If you create a value for this (eg, yourcode), any url to the download.aspx page that includes “&debug=yourcode” will bypass delivery and will show you information about settings, fields, etc. This is very useful as you get familiar with the product.
	reset_key	If you create a value for this (eg, resetcode), any url to the download.aspx page that has both a GUID and “&reset=resetcode” will reset the download count for that customer/item. Note that reset is only available with a valid license.
	page_defs	This is a list of page definitions, separated by “;”. Each definition is the title of a field that contains the page_def.
Page Def		
	Ext=	The file extension that uses this page def. Files that match the extension are assumed to be .PDF format, and

		will have the .ext replaced with .pdf when delivered. This methodology is used so that you can control template usage without requiring any additional settings in the shopping cart module. Just change the .pdf file extension to the one matching the desired template and you're good to go.
	Unit=	Internally, everything is in points. However, it's often difficult to work with, especially when measuring where you want elements to go. The options are: point, cm, mm, inch. Works with most settings except fontsize.
	Lock=	1=lock the pdf from editing; 0=allow editing. Default is 0. Note: <i>while locking should prevent casual modification attempts, there are programs in the wild that can remove these locks (they are part of the pdf spec and not entirely secure).</i>
	Font=	The name of the desired font. Defaults to Helvetica. Note that the fonts are dependent on what is supported by iTextSharp. Available iTextSharp fonts are: Helvetica, Times, Courier. Note: <i>we are working on making more fonts available.</i>
	Fontsize=	In points
	Bold=	1/0
	Italic=	1/0
	Color=	RGB values (RED, GREEN, BLUE) – "color=0,0,255" is blue.
	Date=	Format for date strings; eg "yyyy/MM/dd HH:mm" would yield "2010/05/01 23:21". Uses .Net DateTime.ToString() format.
	Stamp:	Demarks the beginning of a stamp definition for the page. You can have as many stamps as you want (within reason 😊)
Stamp fields		
	Left=	How far from the LEFT edge of the page to place the aligned edge of the text.
	Top=	How far up from the BOTTOM of the page to place the text. While top down would have been nice, this is a function of the iTextSharp component.
	Align=	Text alignment - L, C, R
	Rotation=	In degrees (0-360)
	Text=	The text to stamp. This is a literal string, except that order data can be included by using stamp variables (see section below).
	Date=	Overrides PageDef value
	Font=	Overrides PageDef value
	Fontsize=	Overrides PageDef value
	Bold=	Overrides PageDef value
	Italic=	Overrides PageDef value

Color=	Overrides PageDef value
Stamp text variables	
[FIRSTNAME]	The customer's first name, taken from the Order record.
[LASTNAME]	The customer's last name, taken from the Order record.
[COMPANY]	The customer's company, taken from the Order record.
[ORDER]	The Smith Cart order id, taken from the Order record.
[DETAIL]	The smith Cart order detail id, taken from the Order Detail record.
[GUID]	The detail item guid (from the cart)
[DATE]	The date of the download, formatted as defined in the "date=" option.

The swGateKeeper-Defaults.aspx file contains a sample template called "portrait"

The Download.aspx file

The download.aspx file is the container for swGateKeeper. You can rename the file or place it in the DNN root if you wish. However, please keep in mind the following should you decide to do so.

```
<%@ Page Language="C#" enableSessionState="true" Debug="true" %>
<%@ Import namespace="SwGateKeeper" %>
<% SwGateKeeper swGateKeeper = new SwGateKeeper(this, "g", ""); %>
```

- Line 1 – do not change
- Line 2 – do not change
- Line 3
 - the 2nd parameter ("g") is the same as the "?g=" used in the Smith Product setup. If you change it to something else (eg, guid=) make sure you change it in both places.
 - The 3rd parameter is the path to the swGateKeeper folder from the DNN root (if "" then it's the same folder), which is required in order for the component to find the configuration. For example, if you place the Download.aspx file in the dnn root and the swGateKeeper folder is /swGateKeeper, then use "/swGateKeeper" as the 3rd parameter.

Usage Notes

- Since the file extension is used to select templates, any time a file is delivered with a non-pdf extension you should first check to make sure your page_def extension is set correctly.
- While we have provided a sample portrait template, the landscape and legal templates in the "page_defs" field are not included. You will get a message to this effect if you include a debug= parameter.
- The debug= parameter will fail if used with a guid whose access has expired. Either reset the guid, choose another guid, or remove the "g=" from the url.
- For testing purposes, you can use "Download.aspx?g=1" to get a sample pdf file with a sample stamp data (John Public, Acme Widgets). You can use this sample .pdf to work on developing your own templates.
- Per the above specs, the "top=" option is measured from the bottom of the page.
- Transmitting large PDF files may result in a page timeout. This is not due to stamping (which occurs very quickly), but rather the time it takes to physically transmit the file. Including the following line in the

web.config file can fix this, though since it also extends the page timeout for all pages, you should be careful when using it.

```
<httpRuntime executionTimeout="600" useFullyQualifiedRedirectUrl="true"
maxRequestLength="8192" requestLengthDiskThreshold="8192" />
```

Version History

- 1.0.1 – 5/24/2010
 - PDF stamping
 - Download count/days

Contacting Support

- Please visit the GateKeeper forum at smith-consulting.com.