Datasheet - Sitekit CMS Secure Forms

Document Control	
Registered Office	Sitekit Ltd Sitekit House Broom Place Portree Isle of Skye IV51 9HL
Business Unit	Sitekit.Solutions
Role	Support-Services
Document Type	Manual
Document Author	David Morgan
Document Checked by	Chris Reynolds
Date Checked	05/06/2013
Document Title	Datasheet - Sitekit CMS Secure Forms
Document ID	SKDOC-11-916
Document Version	2.6
Approved Document Version	2.0
Document Approver	David Morgan
Document Status	Approved
Creation Date	15/07/2013
Last Modified Date	23/03/2017
Next Review Date	14/03/2014
Audience	Client
Publisher Rights	© Sitekit Ltd 2013
This document is uncontrolled when printed	

Overview

HTTPS certificates provide both site visitors and administrators with basic security when they are working on a Sitekit web site. But basic security may not be enough when a visitor is submitting sensitive information into one of your web forms, in which case you may want to use a Sitekit CMS Secure Form.

Sitekit CMS Secure Forms are doubly secure. First they employ HTTPS between the visitor's browser and your website form. Secondly they encrypt the submitted data, preventing anyone – even Sitekit Administrators – from seeing the data, unless they hold the decryption key.

Sitekit CMS Secure Forms have two advantages over HTTPS alone. First, they are a core component of Sitekit, and hence available at no extra cost. Secondly, they are extremely secure, since both the page and the submitted data are strongly encrypted.

Communication between the visitor's browser and a secure form on your site are handled via HTTPS using a 256bit EV SSL Certificate. A visitor's submitted form data can only be decrypted using a public/private RSA based key with 1024bit encryption. This key is associated with a single administrative user, rather than a Group.

If a visitor is **no**t submitting sensitive information, then Sitekit CMS Secure Forms probably represent 'overkill', and it would be better to protect selected pages and forms via a normal SSL Certificate, so the form submission is secure but the form data can be stored and retrieved by all administrative users (with the rights) in plain text.

Sitekit CMS Secure Forms are available to both hosted and deployed systems. It is not essential to have an SSL Certificate to use Sitekit CMS Secure Forms, because they are part of Sitekit's core functionality. But without an SSL Certificate, Secure Forms will display something like **https://secure.sitekit.net** in a visitor's title bar, instead of the site's usual domain name. This may be off-putting to visitors, and is not recommended.

If you choose to hold the key locally, rather than letting Sitekit hold it for you, then you must be very careful to keep a secure copy of your key and passphrase. **Without the key and passphrase, submitted encrypted form data is not recoverable.**

Creating and Using Sitekit CMS Secure Forms

As previously mentioned, Sitekit CMS Secure Forms are extremely secure. But whereas using an SSL certificates is straightforward, using Sitekit CMS Secure Forms requires a little more effort. There are two methods of implementing Sitekit Secure Forms.

- Using the secure admin interface to create and associate an encryption key with a specific site user
- Using a Sitekit created desktop utility to create an encryption key and associate it with a specific site user
- Getting Sitekit support to create and encryption key and associate it with a specific user

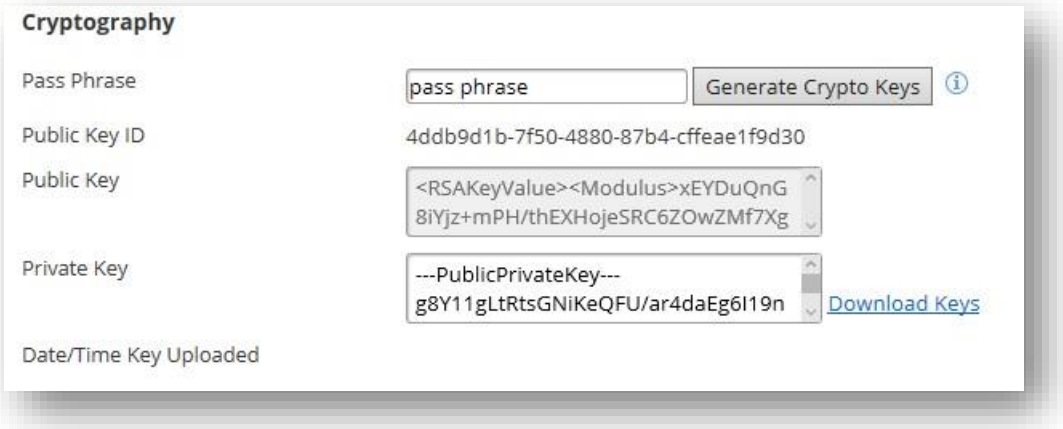
The two methods are described in detail below. For both options, the decryption method is the same and takes place in the admin system with the relevant keys and passphrase

Method 1 – Do it yourself via the admin interface

You can create and associate a key with a specific user via the CMS admin interface:

1. Create a new Admin User in the CMS (Manage… Users…) who will receive the encrypted form submissions. This 'secure' user must have an email address in order to receive the encrypted form submissions. The key can also be associated with an existing user.
2. At the bottom of the new user dialog box there's a **Cryptography** section allowing that user to create and associate keys.
3. Enter a pass phrase and click on the **Generate Crypto Keys**. Pick something which will be easy to remember. This phrase will be needed whenever you decrypt securely submitted form data

4. Your public and private keys have now been created. You need to save your private key. Click on the **download keys** link to save your private key locally. This key file should be stored safely away, somewhere where it cannot be stolen or accidentally deleted. Don't keep it on the pc which created the key in the first place. This is your security backup. This should be kept in a different location from your passphrase. The key file should have the name of the relevant user and a '.key' file extension



5. Click on **Publish** at the top of the dialog box to save your changes and more importantly to associate the key with that user. The interface will change to that shown below with the Date/Time uploaded showing the association timestamp.

6. Create a Sitekit CMS Secure Form. This is simply a case of ticking the **Encrypted** box when creating a new form, and making sure the Notification Email is sent to the email address of the 'secure' user that was configured above.

Method 2 – Using the Sitekit desktop utility

Creating and using a Sitekit CMS Secure Form using local Encryption involves six steps:

1. You must download a copy of the Sitekit Encryption Tool from the Utilities page of the Sitekit Extranet.

2. If you don't already have it installed, you'll need a copy of .net 2 on your pc, in order to use the Sitekit Decryption Tool.

3. You need to create a new User on your site. Make a note of the name and password, because you'll need them later on. It's best to use an email address specific to this form – *MySecureform@myaddress*, for example, rather than the name of an individual. (If you use an individual's address, it's inconvenient should the individual leave, or no longer wants to be responsible for the form).

4. Start the Encryption Tool and use it to create a key which will be used to encrypt/decrypt data in the secure form.

5. Use the Sitekit Encryption Tool to upload the key to your site. The key is associated with the User you created in step 3.

6. Create a new Form. Now that you've uploaded a key to your site, you're offered the option to make the Form a Secure Form.

You now have a working Sitekit CMS Secure Form. All data submitted by visitors using this form is held on your website in encrypted form. You must use the relevant key to read this data.

Each of the six setup steps is now described in detail:

Step 1: Download the Sitekit Encryption Tool

Go onto the Sitekit Extranet and navigate to the Utilities Page. The Encryption Tool is available there as a download. Download the utility and store it safely. Given the high security nature of this utility, it's best to

store it somewhere with controlled access.
https://billing.sitekit.net/SitekitUtilities/SitekitDecryptionTool/publish.htm

Step 2: Download .Net 2

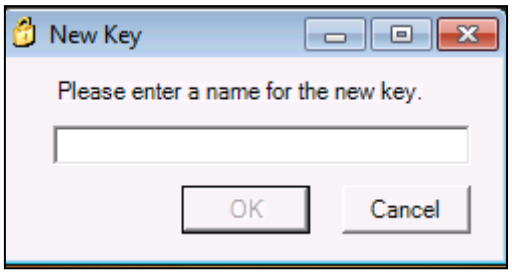
Any pc which you want to run the Encryption Tool must have .NET Framework v2 or later installed. This is available from Microsoft as a FOC download. If you don't already have .NET 2 installed, the Encryption Tool will prompt you to install it.

Step 3: Create a New User on your site

Make a note of the User Name and Password, because you'll need them in a moment or two. This user must have an email address, because the Secure Form will send encrypted form submissions to this email address. You can only select a single email address for the new user – so you might wish to have the email address actually a group address – Secure Form Receipt, say – rather than the email address of an individual. This would allow several people access to the encrypted form submissions. Avoid using an email address for a real individual, who may leave the organisation, causing problems).

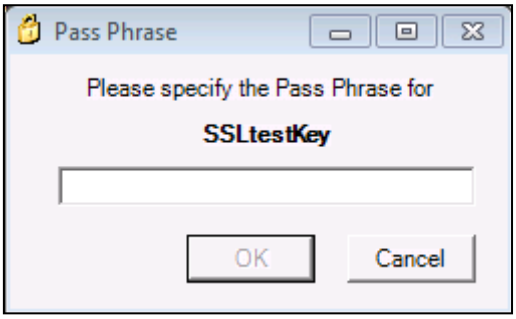
Step 4: Create an Encryption Key

I. Start the Sitekit Encryption Tool, select the Key tab, and click Generate Key. The screen shown below will open. Type in a name for the key. This should be descriptive – the name of the secure form, perhaps.



When you click OK, the Pass Phrase screen opens. Type in a pass phrase. Pick something which will be easy to remember!

II. This phrase will be needed whenever you use the Encryption Utility to read submitted form data! So it's essential that you don't lose it. It's also fairly essential that you keep access to the passphrase limited to authorised personnel, otherwise you frustrate the purpose of encrypting the form data.

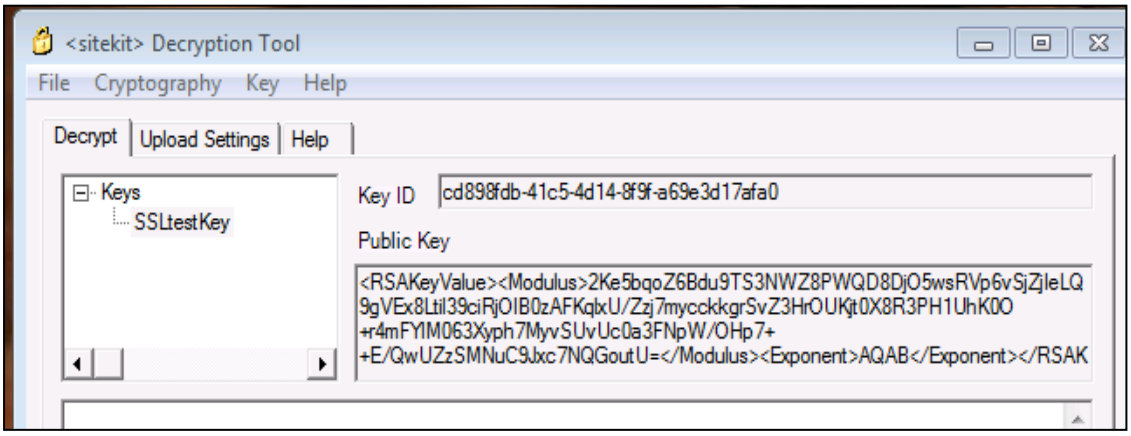


III. Now select **Key>>Export Key**. This will export the key Name and Password to a Key file. This key file should be stored safely away, somewhere where it cannot be stolen or accidentally deleted. Don't keep it on the pc which created the key in the first place! This is your security backup.

Important: The exported Key file does not include the Pass Phrase, which for obvious security reasons must be kept separately.

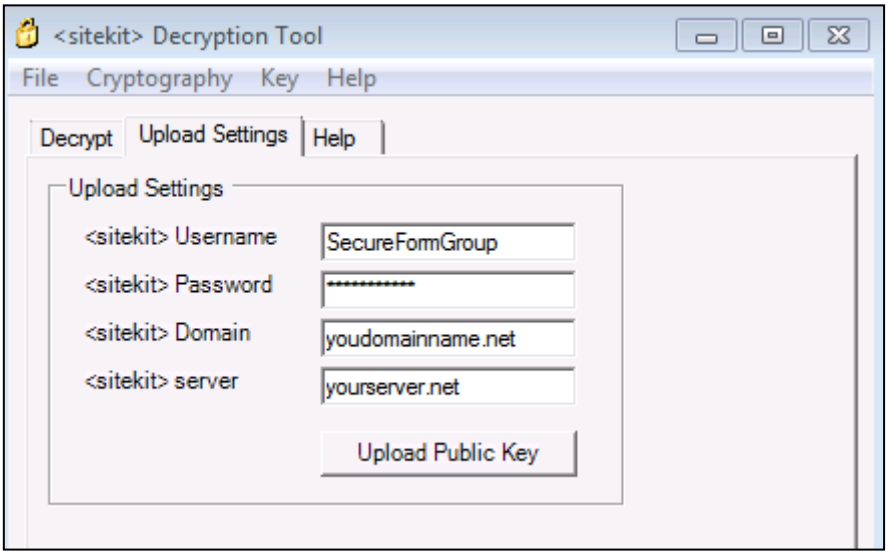
Step 5: Upload the Encryption Key to your site.

I. Click the Key tab and highlight the newly created key. When you do so, a screen similar to the one shown below will open: This shows the key ID and public key.



II. Click the **Key>>Upload Public Key** option. Highlight the key (SSLtestkey, in the graphic above). This opens a screen similar to the one shown below. You have to enter the Sitekit Username and Password that you created in Step 3. Also your Domain Name and Server Name.

III. When you've entered this information, click the Upload Public Key button. This will upload the public key to your site, and associate it with the named User.



Step 6: Create a Sitekit CMS Secure Form

The final step is to create the Secure Form. This process is identical to creating any other form – with one small difference. When creating a secure form, you need to check the **Secure** check box. You must set the Notification Email address to the User associated with the encryption code, as defined in step 5 above.

Note that the desktop encryption tool also contains functionality allowing you to decrypt form submissions. This functionality is deprecated and should not be used. Instead the secure form submissions should only be decrypted from the admin interface as described in the final part of this data sheet

Method 3 – Get Sitekit support to create a key for you.

If you don't want to install and run the Decryption tool locally, you can get Sitekit to create a key for you. If you have a Sitekit Support Contract this service is free of charge, otherwise it is chargeable.

7. Create a new Admin User who will receive the encrypted form submissions. This 'secure' user must have an email address in order to receive the encrypted form submissions.

8. Contact Sitekit, tell them you want an Encryption Code, and give them this newly created User Name and Password. Sitekit will create a new encryption code and associate it with the User. Sitekit will give you your own Private Key and Pass Phrase(??). Store these securely.

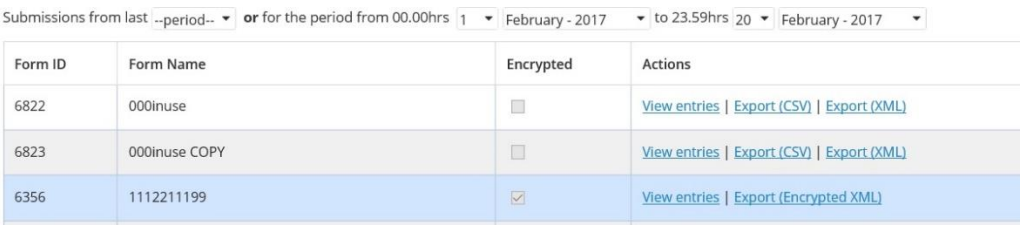
9. Create a Sitekit CMS Secure Form. This is simply a case of ticking the **Encrypted** box when creating a new form, and making sure the Notification Email is sent to the email address of the 'secure' user.

Viewing Data from the Secure Form

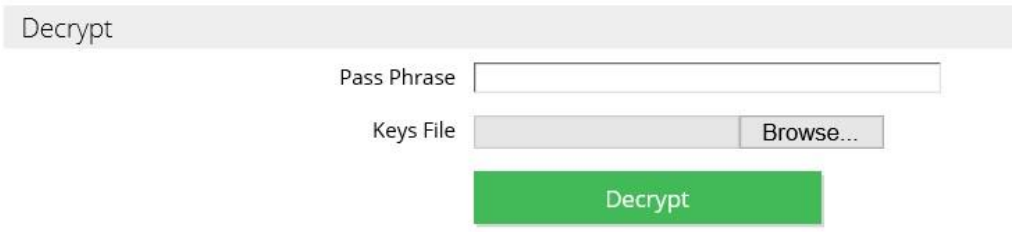
The major difference between a normal form and a Sitekit CMS Secure form is that form submissions are held in an encrypted format. You cannot 'view' form submissions, as you would submissions to a normal form. There are only one way to access the data.

1. When a form submission is made, a notification email in plain text is sent automatically emailed to the Secure User.

2. Inside Sitekit Admin, in the Manage… Forms section you can navigate to the relevant form and click on to 'View entries' to view the data.



3. If the admin system is secured via https the form submission page contains a box at the top for the submission of the passphrase and also a file picker allowing you to navigate to wherever you left the relevant key file. The file is usually named after the associated username with an '.key' file extension.



4. Enter the passphrase and browse to the key, if they match the popup will reload displaying the decrypted form data.