

Symmetric Key Encryption/Decryption Assertion

Introduction

This document is a guide for the Graphical User Interface of the Symmetric Key Encryption Decryption Assertion

Audience

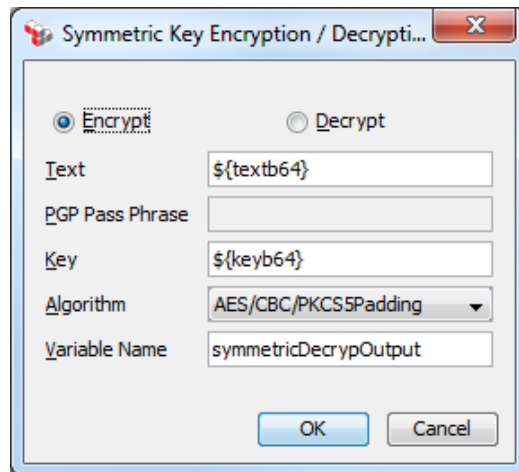
The contents of this document is aimed at users of the Layer 7 Gateway wishing to use this assertion

Version	Changed By	Date Changed
vo.1	R Moshfeghi	11//2011
V1.0	R Moshfeghi	07/25/2012
V1.1	R Moshfeghi	08/09/2012

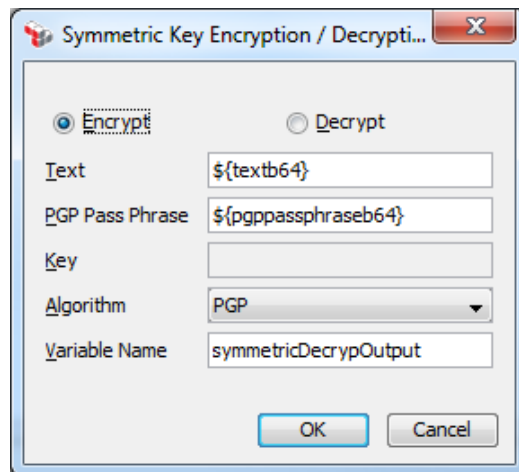
Graphical User Interface

The Graphical User Interface (GUI) of the Symmetric Key Encryption/ Decryption has several different modes depending on the cryptographic algorithm chosen and the cryptography mode.

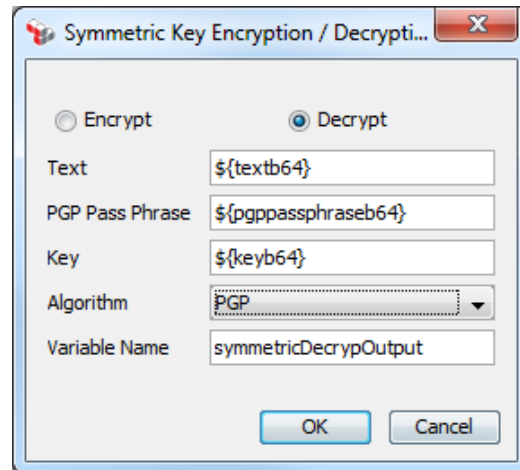
- General Layout:



- PGP Algorithm-Encryption:



- PGP Algorithm-Decryption:



The dialog box is titled "Symmetric Key Encryption / Decrypt...". It has two radio buttons: "Encrypt" (unselected) and "Decrypt" (selected). Below the radio buttons are five input fields: "Text" with value "\${textb64}", "PGP Pass Phrase" with value "\${pgppassphraseb64}", "Key" with value "\${keyb64}", "Algorithm" with a dropdown menu showing "PGP", and "Variable Name" with value "symmetricDecrypOutput". At the bottom are "OK" and "Cancel" buttons.

Input Fields

Input Field	Description
Text	<p>The field that will be encrypted/decrypted.</p> <ul style="list-style-type: none"> • Encryption: plain text • Decryption: cipher text <p>Datatype: Base 64 encoded String</p> <p>Can be a literal or a context variable.</p>
PGP Pass Phrase	<p>Pass Phrase utilized during PGP based encryption and decryption.</p> <p>This field is only available when PGP is chosen as the algorithm.</p> <p>Note: It is highly recommended to store the PGP Pass Phrase in the Gateway's Password Store which can be accessed via the <i>Manage Store Passwords</i> task. The contents of the Store can be accessed via context variables.</p>
Key	<p>The symmetric key that will be used in the encryption/decryption process. This text field is available in all modes except when PGP and the Encryption radio button are chosen together.</p> <p>Characteristics of the key dictate characteristics of the Algorithm chosen:</p> <p>AES:</p> <ul style="list-style-type: none"> • 128 bits chooses AES128 • 192 bits chooses AES192 • 256 bits chooses AES256 <p>DES and DESede:</p> <ul style="list-style-type: none"> • 64 bits <p>PGP:</p> <ul style="list-style-type: none"> • Only available during the Decryption process and maps to a PGP Private Key in the format of:

Input Field	Description
	<p>-----BEGIN PGP PRIVATE KEY BLOCK-----</p> <p>.....</p> <p>-----END PGP PRIVATE KEY BLOCK-----</p> <p>Datatype: Base 64 encoded String</p> <p>Can be a literal or a context variable</p> <p>Note: It is highly recommended to store the Key in the Gateway's Password Store which can be accessed via the <i>Manage Store Passwords</i> task. The contents of the Store can be accessed via context variables.</p>
Algorithm	<p>AES/CBC/PKCS5Padding</p> <ul style="list-style-type: none"> AES algorithm (either 128, 192 or 256 depending on the size of the key) with CBC block mode and PKCS5Padding <p>DES/CBC/PKCS5Padding</p> <ul style="list-style-type: none"> DES algorithm with CBC block mode and PKCS5Padding <p>DESede/CBC/PKCS5Padding</p> <ul style="list-style-type: none"> Triple DES algorithm with CBC block mode and PKCS5Padding <p>PGP</p> <ul style="list-style-type: none"> Encryption: <ul style="list-style-type: none"> Key Generation: SHA-512 (Iterated and Salted) Encryption: AES 256 bit Integrity: enabled and using SHA-1 algorithm ASCII Armor: false Decryption: <ul style="list-style-type: none"> If the user specifies a Key, the Key is treated as a PGP Private Key and it along with the PGP Pass Phrase are utilized to decrypt the Text. If the user only specifies a PGP Pass Phrase, only it used to decrypt the Text. The assumption is that the Private Key is encrypted along with the Text. If the integrity bit has been enabled on the encrypted text and it fails verification during the decryption process, the entire process will fail.
Variable Name	<p>The name of the context variable that will contain the output of the assertion</p> <p>Default: symmetricEncrypDecrypOutput</p> <p>Literal.</p>

Output Fields

- Context Variable with the name specified for "Output Variable Name"
- Datatype: Base 64 Encoded String

General Algorithm

Encryption:

- cipher text output = encrypt(algorithm, text, key)
 - output variable will have the name "Variable Name"

Decryption:

- plain text output = decrypt(algorithm, text, key)
 - output variable will have the name "Variable Name"