AES-256-CBC Webhook Decoder Documentation Overview

This PHP script decrypts AES-256-CBC–encrypted webhook responses. It uses from Business a your **private key** and the website's **public key** to perform the decryption. The decrypted content is parsed from **$_REQUEST['encryption_data']**, which contains URL-encoded key-value pairs representing the webhook payload.

Configuration

```
- $website_public_key: Should be your public_key
- $private_key: Should be your private_key
```

Decryption Function

```
function data_decodef($string,$private_key,
$website_public_key) {   //decode string which encode via
AES-256-CBC hash with private_key and token
    $output = false;
    $encrypt_method = "AES-256-CBC";    //encrypt method
    $iv = substr( hash( 'sha256', $website_public_key ), 0,
16 );
    $output = openssl_decrypt( base64_decode( $string ),
$encrypt_method, $private_key, 0, $iv );
    return $output;
}
```

Webhook Response Decryption

Replace with your actual webhook response from $_REQUEST['encryption_data']

```
$webhook_response_encrypted = @$_REQUEST['encryption_data'];
$decoded = data_decodef($webhook_response_encrypted,
$private_key, $website_public_key);
```

Parsing Decoded Data

```
parse_str($decoded, $decoded_array);
print_r($decoded_array)
```

PHP Code

```
<?php
// Decode AES-256-CBC encrypted webhook response

$website_public_key = "MjcyXzEwXzIwMjQwNjA0MTcwMzIw"; //
Should be your public_key
$private_key = "MjcyXzIwMjMwOTI2MTIxMzUx";       // Should
be your private_key

// Enable error reporting
error_reporting(E_ALL);
ini_set('display_errors', '1');
ini_set('max_execution_time', 0);

/**
 * Decrypt AES-256-CBC encrypted string
 *
 * @param string $string Encrypted string
 * @param string $private_key Private key for decryption
 * @param string $public_key Public key to derive IV
 * @return string|false Decrypted string or false on failure
 */
function data_decodef($string,$private_key,
$website_public_key) {
    $output = false;
    $encrypt_method = "AES-256-CBC";    //encrypt method
    $iv = substr( hash( 'sha256', $website_public_key ), 0,
16 );
    $output = openssl_decrypt( base64_decode( $string ),
$encrypt_method, $private_key, 0, $iv );
    return $output;
}


// Replace with your actual webhook response
$webhook_response_encrypted =
"cS92L3ZIRVhXZmVIZnVqaThtWHhmWUdwMVN5Z2Z0ZFpnV2ExTXdLNDA0RDl
4ZHNNd1ViL3FtQUY5TnhMaWpLckRxNDZ6RjJEVTFYUnVMU0NPVXVrYjQ3LzF
SUjVwSjVvYytDdm04MUF1Q1F5eVV3Z3daWjQ3c3huMTdqK0ExVlB6a0xXaE8
1dG8veklGbXErWU5hUG9lbUplNlFPSHJDWFlGelhKVk1LWkRwbmZKZkVhQm0
rekNtREhuazk2ZjIzbWIyM1ErSi9uRFEvZzdZR0Z1TnlQaXJtMndmZDZjZVJ
LTDY1cm5EZmVuQTI2cjkxRVpjWlBCQi9HejlpWnlFSWxJU0ptOXQwSHEwdWJ
NS0ZyL3Y0dmtjKzBGTGZ2VS9Vd3Rtc2taNlNyOXVjNVNJVjdFT255NzJXcnZ
WcVdhaURGMGRBNTZNQ1hXaW5XS1dDVW1hTG1xR01sMWlSbmwvNDJkcTNxcEZ
YeS9YQmdVN3ErQnNrcVFTRWhkZW4xWDFQQlJYSlNiRnZHTkxRZnA1dEQ5Snp
5UUE2bTgvaWR0OTRvRko1UzZ3aDZGR2VPaGVkTnZFUkFxdGhzQ3lUWm9PYnh
```

```
rMUZtazlNMHIxVEFQZDdNamVXcFVwcnA5akpNUVlucUs1ckxDVG0vb0k9";

echo "<h2>Webhook Response (Encrypted)</h2>";
echo "<pre>$webhook_response_encrypted</pre>";

// Decode the webhook response
$decoded = data_decodef($webhook_response_encrypted,
$private_key, $website_public_key);

echo "<h3>Decoded Webhook Response (Raw & Subquery Pram)</
h3>";
echo "<pre>$decoded</pre>";

// Decode JSON
parse_str($decoded,$decoded_array);

echo "<h3>Decoded Webhook Response (Array)</h3>";
echo "<pre>";
print_r($decoded_array);
echo "</pre>";


?>
```