

# Zend\_Crypt\_Xml - Steven George (Deakin University)

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

<ac:macro ac:name="unmigrated-inline-wiki-markup"><ac:plain-text-body><![CDATA[

## Zend Framework: Zend\_Crypt\_Xml Component Proposal

<b>Proposed Component Name</b>	Zend_Crypt_Xml
<b>Developer Notes</b>	<a href="http://framework.zend.com/wiki/display/ZFDEV/Zend_Crypt_Xml">http://framework.zend.com/wiki/display/ZFDEV/Zend_Crypt_Xml</a>
<b>Proposers</b>	Steven George
<b>Zend Liaison</b>	TBD
<b>Revision</b>	1.0 - 17 July 2012: Initial Draft. (wiki revision: 7)

## Table of Contents

1. Overview
2. References
3. Component Requirements, Constraints, and Acceptance Criteria
4. Dependencies on Other Framework Components
5. Theory of Operation
6. Milestones / Tasks
7. Class Index
8. Use Cases
  - UC-01 Encrypting XML
  - UC-02 Decrypting XML
9. Class Skeletons

### 1. Overview

In a world where security is paramount, there is a need to secure sensitive data that is stored with PHP applications on the file system. Passwords and other sensitive type data should never be stored in plain text.

Zend\_Crypt\_Xml is a class that allows encryption of nodes of an XML document. This would be typically used to encrypt credentials in a configuration file. This class will then allow this encrypted data to be decrypted at run-time (in memory). This means that plain-text passwords are never stored anywhere, including on the filesystem or in revision control systems.

### 2. References

- (none)

### 3. Component Requirements, Constraints, and Acceptance Criteria

- This component will encrypt sections of an xml document.

- This component will rely on other components of the Zend\_Crypt package to aid the encryption activity.
- This component will require the developer to generate a public / private key set.
- This component will require the developer to specify an encryption method.
- We recommend building a simple web UI on top of this component to facilitate the encryption process.

## 4. Dependencies on Other Framework Components

- Zend\_Crypt

## 5. Theory of Operation

The component is instantiated by passing an instance of Zend\_Crypt\_\* that supports two-way encryption. Once this is instantiated, the developer can pass an xml string to the "encrypt" or "decrypt" methods of Zend\_Crypt\_Xml.

The "encrypt" method will search the xml document for nodes that contain the attribute 'encrypt="true"'. Once found, the contents of this node will be encrypted using the given algorithm.

A number of elements will be added to the xml document:

- "EncryptionMethod" - Outlines the encryption method that was used
- "KeyInfo" - Provides the key
- "CipherData" - Contains the data package
- "EncryptedData" - Contains the encrypted data

The "decrypt" method will search the xml document for encrypted nodes. Once found, the method will read the encryption method and key and decrypt using the relevant algorithm.

## 6. Milestones / Tasks

- Milestone 1: [DONE]Proposal
- Milestone 2: Working prototype checked into the incubator
- Milestone 3: Unit tests exist, work, and are checked into SVN.
- Milestone 4: Initial documentation exists.

## 7. Class Index

- Zend\_Crypt\_Xml

## 8. Use Cases

### UC-01 Encrypting XML

BEFORE:  
=====

```
<?xml version="1.0"?>
<configdata>
  <production>
    <credentials encrypt="true">
      <username>bob</username>
      <password>pass123</password>
    </credentials>
  </production>
</configdata>
```

```
$crypt = new Zend_Crypt_Xml(new Zend_Crypt_Rsa('/path/to/privatekey.pem'));
$encryptedData = $crypt->encrypt($xml);
```

AFTER:  
=====

```
<?xml version="1.0"?>
<configdata>
  <production>
    <EncryptedData><EncryptionMethod Algorithm="RSA"/><KeyInfo
xmlns="http://www.w3.org/2000/09/xmldsig#"><EncryptedKey
xmlns="http://www.w3.org/2001/04/xmlenc#"><CipherData><CipherValue>LS0tLS1CRUdJTiBQVUJMSUMgS0VZLS0tLS0KTU1HZ
</production>
</configdata>

*/
```

## UC-02 Decrypting XML

```
$crypt = new Zend_Crypt_Xml(new Zend_Crypt_Rsa('/path/to/privatekey.pem'));
$xml = $crypt->decrypt($encryptedData);
```

## 9. Class Skeletons

```
class Zend_Crypt_Xml
{
    /**
     * @param Zend_Crypt $encryptionProvider The class to use for encryption
     */
    public function __construct( $encryptionProvider = null )
    {

    }

    /**
     * @param string $xml The xml document as a string
     *
     * @return string The encrypted xml document as a string, or false if no data found to encrypt
     */
    public function encrypt($xml)
    {

    }

    /**
     * @param string $xml The xml document as a string
     *
     * @return string The decrypted xml document as a string, or false if no encrypted data found
     */
    public function decrypt($xml)
    {

    }
}
```

```
]]></ac:plain-text-body></ac:macro>
]]></ac:plain-text-body></ac:macro>
```