Pubkey Encrypt: Architecture Document

Project Objectives:

- Provide an easy-to-use website security enhancement module for administrators
- Use ownCloud's Data Encryption Model in the context of Drupal
- Protect websites' Data at Rest even in the cases of data breaches

Project Goal:

- Add support for Public key based encryption mechanism into the Field Encrypt module

Project Dependencies

- Field API - allows custom fields to be attached with Drupal entities
- Key module - provides API for managing keys and key-based operations in Encrypt
- Encrypt module - provides API for managing encryption plugins and operations
- Field Encrypt module - adds options to encrypt field values
- OpenSSL - provides protocols for the symmetric and asymmetric encryption

How the Module will Integrate with its Dependencies:

Field API: The module will register three additional fields with the User entity

- Public key (string)
- Private key (string)
- Protected (bool)

In this way each user will get a Public/Private key pair assigned to him. The Protected bool will tell whether the Private key for a user has been saved in its original form or has it been further encrypted with that user's credentials and then saved.

It should be noted here that the Public/Private keys assigned to users will never ever change. And the module will not allow anyone to configure these fields from the UI either.

Key API: The module will implement a Key Provider plugin such it will not store the key values in their original format. But instead:

:

- Upon key storage, it'll encrypt the Role key with the Public keys of all users from the specified role. Then the key provider will store these multiple encrypted copies (i.e. one

encrypted copy per user) of a single key in the Drupal configuration system. Then the Role key will be discarded.
- Upon key retrieval, it'll decrypt one of the multiple encrypted key copies with the Private key of the logged in user. Then the key provider will return the decrypted Role key and Role. So the Role key used for actually encrypting the data won't get stored in the server in it's original format.

Encrypt API: The module will define multiple encryption profiles to be used for encrypting field data. Each role defined in a website will have an associated encryption profile for it, so to allow administrators chose the most appropriate encryption profile depending upon their data. For example, an article body visible only to people having the "Premium Users" role should be encrypted with the corresponding "Premium Users" encryption profile.

Field Encrypt: The module will leverage the Field Encrypt module for the actual encryption/decryption events on field data.

Architecture of the Module:

The uniqueness of the module's functionality comes from the fact that it will use user-credentials in the data encryption phases. So, the Data at Rest will remain protected even in cases of data breaches. By default, the module will add support for Password-based public key encryption which means users' passwords will be used during both encryption and decryption of data. But it's not a far fetched assumption to say that an organization might want some other user-credentials (like PIN etc.) to be used. Therefore, we've decided to make:

- **A pluggable system for providing user-credentials:** The plugin would take a user ID as the input and would return the user-credentials as the output. Then it's the duty of the module to use the provided user-credentials during data encryption and decryption phases.
- **A pluggable system for generating Public/Private key pairs.**

Here are all the events and their associated responses which the module will trigger:

- **Module Installation:** Initialization of the additional three fields for all users
- **User registration:** Initialization of the additional three fields for a new user
- **User first-time login:** Protection of the Private key field with the user credentials
- **User credentials change:** Modification of the Private key field so to protect it with new user credentials. It should also be noted here that a credentials change will not imply the need of re-encrypting data again.
- **New role addition:** Creation of the corresponding Encryption profile