'attributes' => [

ActiveRecord::EVENT\_AFTER\_INSERT => 'encrypted\_field', ActiveRecord::EVENT\_AFTER\_UPDATE => 'encrypted\_field',

],

'value' => function ($event) {

return $event->sender->encrypted\_field\_temp;

},

],

[

'class' => AttributeBehavior::className(),

'attributes' => [

ActiveRecord::EVENT\_AFTER\_FIND => 'encrypted\_field',

],

'value' => function ($event) {

return Yii::$app->security->decryptByKey( $event->sender->encrypted\_field,

Yii::$app->params['key']

);

},

],

];

}

3. Add controllers/CryptoController.php:

<?php

namespace app\controllers;

use app\models\Order; use Yii;

use yii\db\Query;

use yii\helpers\ArrayHelper;

use yii\helpers\Html;

use yii\helpers\VarDumper;

use yii\web\Controller;

/\*\*

* Class CryptoController.
* @package app\controllers \*/

class CryptoController extends Controller {

public function actionTest()

{

$newOrder = new Order();

$newOrder->client = "Alex";

$newOrder->total = 100;

$newOrder->encrypted\_field = 'very-secret-info';

$newOrder->save();

$findOrder = Order::findOne($newOrder->id);

return $this->renderContent(Html::ul([

'New model: ' . VarDumper::dumpAsString($newOrder->attributes),

'Find model: ' . VarDumper::dumpAsString($findOrder->attributes)

]));

}

public function actionRaw()

{

$row = (new Query())->from('order')

->where(['client' => 'Alex'])

->one();