Cryptography

Non-Visible component

Category	Requires	Version
Utilities	API 19, Android 4.4 - 4.4.4 KitKat	2

Overview

A non-visible component that encrypts and decrypts data using a variety of techniques.

Methods

AES-128 Decode

Returns: *Text*

Decodes the given hash using the given key through AES-128. If any exception occurs, returns empty string.

Params	
AES-128 Hash	Text

AES-128 Encode

Returns: *Text*

Encodes the given string using the given key through AES-128. If any exception occurs, returns empty string.

Params	
input Text	Text

AES-256 Decode

Returns: *Text*

Decodes the given hash using the given key through AES-256. If there are any exceptions, returns empty string

Params	
AES-256 Hash	Text

AES-256 Encode

Returns: *Text*

Encodes the given string using the given key through AES-256. If there are any exceptions, returns empty string

Params	
input Text	Text

Base64 Decode

Returns: *Text*

Decodes the given hash using Base64

Params	
base64 Hash	Text

Base64 Encode

Returns: *Text*

Encodes the given string using Base64

Params	
input Text	Text

BCrypt Create Hash

Returns: *Text*

Generates a hash using BCrypt

Params	
input Text	Text
salt	Text

BCrypt Generate Salt

Returns: *Text*

Generates a salt usable for hashing with BCrypt

BCrypt Validate Password

Returns: *Boolean*

Verifies if the input password is the same one as the correct hashed password using BCrypt algorithm

Params	
input Text	Text
correct Hash	Text

Enigma Parser

Returns: *Text*

Encrypts or decrypts the given string simulating an Enigma machine. Rotors can go from 1 to 8, but they cannot be repeated. Reflector can be B, C or 0 if none. Plugboard is a list with sub-list of two items containing a character each one, which replace the first character with the second one.

Params	
input Text	Text
rotor 1	Number
rotor 2	Number
rotor 3	Number
reflector	Text
plugboard	List

Generate AES 128 Key

Returns: *Text*

Generates a secure random AES 128 key

Generate AES 256 Key

Returns: *Text*

Generates a secure random AES 256 key

MD5 Create Hash

Returns: *Text*

Generates a MD5 hash

Params	
input Text	Text

PBKDF2 Create Hash

Returns: *Text*

Generates a hash using PBKDF2

Params	
input Text	Text

PBKDF2 Validate Hash

Returns: *Boolean*

Verifies if the input password is the same one as the correct hashed password using PBKDF2 algorithm

Params	
input Text	Text
correct Hash	Text

SHA-1 Generate Hash

Returns: *Text*

Generates a hashed SHA-1 string

Params	
input Text	Text

SHA-224 Generate Hash

Returns: *Text*

Generates a hashed SHA-224 string

Params	
input Text	Text

SHA-256 Generate Hash

Returns: *Text*

Generates a hashed SHA-256 string

Params	
input Text	Text

SHA-384 Generate Hash

Returns: *Text*

Generates a hashed SHA-384 string

Params	
input Text	Text

SHA-512 Generate Hash

Returns: *Text*

Generates a hashed SHA-512 string

Params	
input Text	Text

TripleDES Decode

Returns: *Text*

Decodes the given hash using the given key through TripleDES

Params	
tripleDES Hash	Text

TripleDES Encode

Returns: *Text*

Encodes the given string using the given key through TripleDES

Params	
input Text	Text

Properties

AES-128 Key

Text — Read Write - Designer Blocks

Set the AES-128 Key

AES-256 Key

Text — Read Write - Designer Blocks

Set the AES-256 Key

BCrypt Salt Size

Number Default: *10* — Read Write - Designer Blocks

Set the BCrypt Salt Size

PBKDF2 Hash Byte Size

Number Default: *18* — Read Write - Designer Blocks

Set the PBKDF2 Hash Byte Size

PBKDF2 Iterations Number

Number Default: *64000* — Read Write - Designer Blocks

Set the PBKDF2 number of Iterations

PBKDF2 Salt Byte Size

Number Default: *24* — Read Write - Designer Blocks

Set the PBKDF2 Salt Byte Size

TripleDES Key

Text — Read Write - Designer Blocks

Set the TripleDES Key

Last update: January 26, 2020