



## Encrypted String is wrong in swift #774

New issue

 Closed

BKudale opened this issue on 6 Nov 2019 · 12 comments

 BKudale commented on 6 Nov 2019 · edited

Describe the bug

Reproduce

Steps to reproduce the behavior:

My Code

```
extension String {
    func aesEncrypt() throws -> String {
        let secreteKey = getSecreteKey()
        let iv : Array<UInt8> = "0000000000000000".utf8.map { $0 }
        guard let data = self.data(using: .utf8) else { return "" }
        let encrypted = try AES(key: secreteKey, blockMode: CBC(iv: iv), padding: .pkcs5).encrypt([UInt8](data))
        let encryptedData = Data(encrypted)
        return encryptedData.base64EncodedString()
    }

    func aesDecrypt() throws -> String {
        let secreteKey = getSecreteKey()
        let iv : Array<UInt8> = "0000000000000000".utf8.map { $0 }
        guard let data = Data(base64Encoded: self) else { return "" }
        let decrypted = try AES(key: secreteKey, blockMode: CBC(iv: iv), padding: .pkcs5).decrypt([UInt8](data))
        let a = decrypted[0]
        let b = decrypted[1]
        let c = decrypted[2]
        let d = decrypted[3]
        let str1 = "\(a)\(b)\(c)\(d)"
        return str1
    }

    func getSecreteKey()-> Array<UInt8> {
        let password: Array<UInt8> = KEY.utf8.map { $0 }
        let salt: Array<UInt8> = SALT.utf8.map { $0 }
        let value = try! PKCS5.PBKDF2(password: password, salt: salt, iterations: 65536, keyLength: 32, variant: .sha256).
        return value
    }
}
```

From above code i decrypted the message sent from server successfully

1. Server Encrypted message - i0oo2F6tAwvHG5OI1YX86A== . After decrypt - 6666 successfully decrypted
2. Tried to encrypt "6666" using above func got output - k1AwAbhS7IgXAPgwDVEBVg==

Which is different . Please suggest what is wrong in above code , Thank you.

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone


No milestone


Linked pull requests

Successfully merging a pull request may close this issue.

None yet

4 participants




 krzyzanowskim commented on 6 Nov 2019

Owner

let iv : Array = "0000000000000000".utf8.map { \$0 }

most likely this is not what you think it is. This is [48, 48, 48, 48, 48, 48, 48, 48, 48, 48, 48, 48, 48, 48, 48, 48], not array of zeroes.


 krzyzanowskim commented on 6 Nov 2019

Owner

let data = self.data(using: .utf8)


this won't encrypt 6666 , it will encrypt "6666" ( a string) - those are different values

  krzyzanowskim closed this on 6 Nov 2019

 BKudale commented on 6 Nov 2019


Author

yes i am encrypting the string "6666" but getting output wrong , Please suggest

 krzyzanowskim commented on 6 Nov 2019

Owner

encrypted 6666 and "6666" has different output. i0oo2F6tAwvHG5OI1YX86A== is for 6666, while k1AwAbhS7IgXAPgwDVEBVg== is for "6666". This is as expected.


 BKudale commented on 6 Nov 2019 · edited by krzyzanowskim

Author

Code updated , converted input string to int but still getting the same result

```
func aesEncrypt() throws -> String
{
    let secreteKey = getSecreteKey()
    let iv : Array<UInt8> = "0000000000000000".utf8.map { $0 }
    let a:Int? = Int(self)
    let strBytes = getAsciiCodesOfDigits(a)
    let encrypted = try AES(key: secreteKey, blockMode: CBC(iv: iv), padding: .pkcs5).encrypt(strBytes)
    let encryptedData = Data(encrypted)
    return encryptedData.base64EncodedString()
}

func getAsciiCodesOfDigits(_ n: Int)->[UInt8]{
    return String(n).unicodeScalars.map{UInt8($0.value)}
}
```

 BKudale commented on 6 Nov 2019 · edited

Author

encrypted 6666 and "6666" has different output. i0oo2F6tAwvHG5OI1YX86A== is for 6666, while k1AwAbhS7IgXAPgwDVEBVg== is for "6666". This is as expected.

@krzyzanowskim Checked with backend code output i0oo2F6tAwvHG5OI1YX86A== is for string "6666" .


Also same code in android is working fine for string "6666" which generating the same output 'iOoo2F6tAwvHG5OI1YX86A='

It is not the datatype issue , Please check my code again , something must be wrong

extension String {
 func aesEncrypt() throws -> String {
 let secreteKey = getSecreteKey()
 let secreteKey = getSecreteKey()
 let iv : Array = "0000000000000000".utf8.map { \$0 }
 guard let data = self.data(using: .utf8) else { return "" }
 let encrypted = try AES(key: secreteKey, blockMode: CBC(iv: iv), padding: .pkcs5).encrypt(UInt8)
 let encryptedData = Data(encrypted)
 return encryptedData.base64EncodedString()
 }

 func aesDecrypt() throws -> String {
 let secreteKey = getSecreteKey()
 let iv : Array<UInt8> = "0000000000000000".utf8.map { \$0 }
 guard let data = Data(base64Encoded: self) else { return "" }
 let decrypted = try AES(key: secreteKey, blockMode: CBC(iv: iv), padding: .pkcs5).decrypt([UInt8](data))
 let a = decrypted[0]
 let b = decrypted[1]
 let c = decrypted[2]
 let d = decrypted[3]
 let str1 = "\(a)\(b)\(c)\(d)"
 return str1
 }

 func getSecreteKey()-> Array<UInt8> {
 let password: Array<UInt8> = KEY.utf8.map { \$0 }
 let salt: Array<UInt8> = SALT.utf8.map { \$0 }
 let value = try! PKCS5.PBKDF2(password: password, salt: salt, iterations: 65536, keyLength: 32, variant: .sha256).calc
 return value
 }
}

 krzyzanowskim commented on 6 Nov 2019

Owner

Checked with backend code output iOoo2F6tAwvHG5OI1YX86A== is for string "6666" .


that doesn't make sense, you wouldn't decrypt that with the function you've built

this

let a = decrypted[0]
let b = decrypted[1]
let c = decrypted[2]
let d = decrypted[3]


is for [6,6,6,6], not "6666", not even 6666

I didn't run the code, I just use what you provided.

 BKudale commented on 6 Nov 2019

Author

@krzyzanowskim ok please suggest what can i do to match the encrypted code 'iOoo2F6tAwvHG5OI1YX86A==' for string "6666" (as it is entered by user) . can i convert it to int ? then pass bytes array of converted int to encrypt func .

 BKudale commented on 6 Nov 2019

Author

Checked with backend code output iOoo2F6tAwvHG5OI1YX86A== is for string "6666" .

that doesn't make sense, you wouldn't decrypt that with the function you've built


this

let a = decrypted[0]
let b = decrypted[1]
let c = decrypted[2]
let d = decrypted[3]

is for [6,6,6,6], not "6666", not even 6666

I didn't run the code, I just use what you provided.


@krzyzanowskim After decrypt we got byte array [6,6,6,6,60,60,60,60,60,60,60,60,60,60,60,60] which contain ascii values which is not an int or string.

 krzyzanowskim commented on 6 Nov 2019

Owner

You have to figure out correct input data and match it with whatever is done on the server. The easiest approach would be to deal with bytes (UInt8) on both sides. Your problem is most likely due do implicit conversions and/or encodings on the way.

Sorry, I can't help you more here.


 sagarauiszz commented on 1 Feb

Issue has been resolved. Please check following code.

```
func getSecreteKey()-> Array<UInt8> {
    let password: Array<UInt8> = plaintext.utf8.map { $0 }
    let salt: Array<UInt8> = saltKey.utf8.map { $0 }
    let value = try! PKCS5.PBKDF2(password: password, salt: salt, iterations: 65536, keyLength: 32, variant: .sha256).calc
    return value
}

func encrypt(content: String) throws -> String {
    let secreteKey = getSecreteKey()
    let iv : Array<UInt8> = "0000000000000000".utf8.map { $0 }
    guard let data = content.data(using: .utf8) else { return "" }
    let encrypted = try AES(key: secreteKey, blockMode: CBC(iv: iv), padding: .pkcs5).encrypt([UInt8](data))
    let encryptedData = Data(encrypted)
    return encryptedData.base64EncodedString()
}

func decrypt(content: String) throws -> String {
    let secreteKey = getSecreteKey()
    let iv : Array<UInt8> = "0000000000000000".utf8.map { $0 }
    guard let data = Data(base64Encoded: content) else { return "" }
    let decrypted = try AES(key: secreteKey, blockMode: CBC(iv: iv), padding: .pkcs5).decrypt([UInt8](data))
    let str1 = Data(decrypted)
    return String(data: str1, encoding: .utf8) ?? "Empty"
}
```

 bovillaios commented on 7 May · edited


@sagarauiszz, Is the above code working. I am trying it now. It is not working. Can you plz help me out.

let passwordString = "3dlPm7jHS4uGzmKvFsXl0UJ7y8QgCdMwMctn+3AAqI="
let saltString = "6t30TEKH3tF8AvKtsD9TVASyU8GdM47vKBLpqT13I="

print(try! encrypt(content: "123456"))
print(try! decrypt(content: encrypt(content: "123456")))

Sign up for free

 to join this conversation on GitHub. Already have an account? [Sign in to comment](#)

 © 2020 GitHub, Inc.

[Terms](#) [Privacy](#) [Security](#) [Status](#) [Help](#) [Contact GitHub](#) [Pricing](#) [API](#) [Training](#) [Blog](#) [About](#)