datatheorem VIEW IN PORTAL

Prevent Keychain Items on 3rd Party Backups (iTunes/iCloud)

CultureNEXT - PreProd 1.0.0 ID S001933 Severity: Medium No Priority ATT&CK

The following items stored by the App in the Keychain are exported to backups:

OPEN since 4/13/2021 Application Tag: *com.puswoosh.attributesPublicKey* Key Class: *kSecAttrKeyClassPublic*

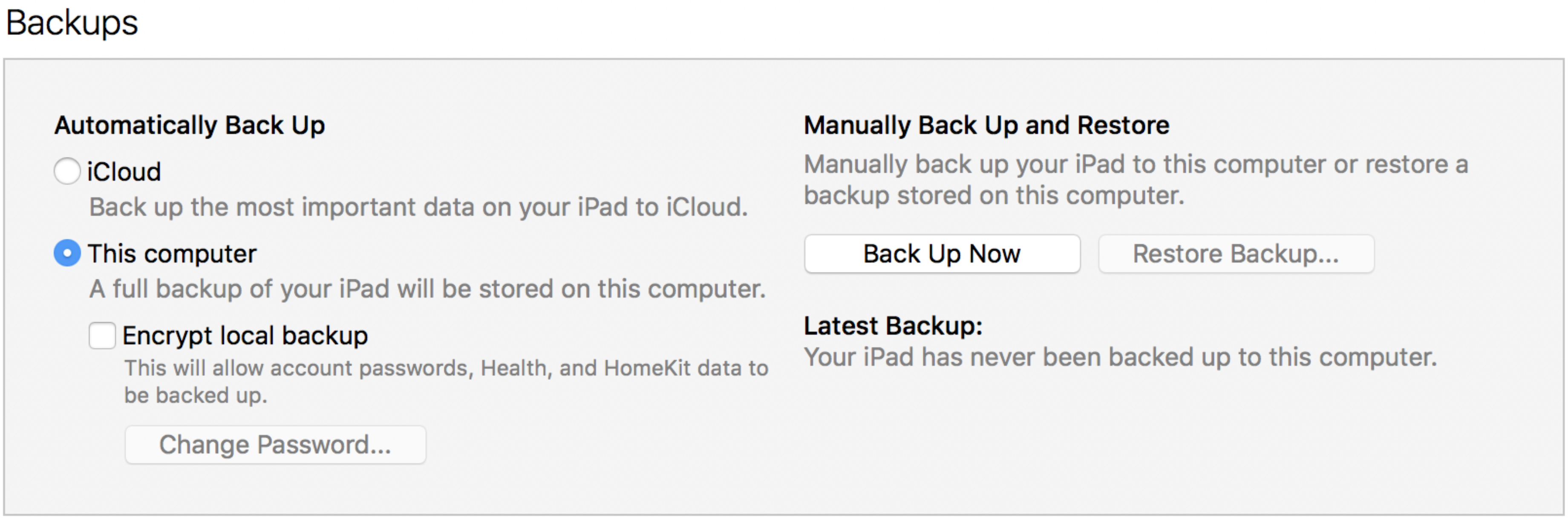
DESCRIPTION

When adding these items to the Keychain, the App does not explicitly exclude the items from device backups, which will allow sensitive data to get exported to iTunes and iCloud backups.

Having sensitive data sent to iCloud exposes it to both Apple, Inc. and an attacker with the ability to compromise the user's iCloud account (similarly to the iCloud celebrity hack, described in https://en.wikipedia.org/wiki/ICloud_leaks_of_celebrity_photos).

Additionally, exporting data to iTunes backups allows an attacker with physical access to the device and its passcode to use the iTunes encrypted backup functionality to extract all the Keychain items (that are flagged for backups) from the device.

SCREENSHOT



Backups

Automatically Back Up

○ iCloud

Back up the most important data on your iPad to iCloud.

● This computer

A full backup of your iPad will be stored on this computer.

☐ Encrypt local backup

This will allow account passwords, Health, and HomeKit data to be backed up.

Manually Back Up and Restore

Manually back up your iPad to this computer or restore a backup stored on this computer.

Latest Backup:

Your iPad has never been backed up to this computer.

RECOMMENDATION

In order to prevent Keychain data from being migrated to iTunes and iCloud backups, explicitly set a `ThisDeviceOnly` accessibility class (such as `kSecAttrAccessibleWhenUnlockedThisDeviceOnly`) for all Keychain items. More information about such accessibility classes is available at https://developer.apple.com/library/ios/documentation/security/Reference/keychainservices/Reference/reference.html.

Additionally, because the official iOS Keychain APIs (including `SecItemAdd()` and `SecItemCopyMatching()`) are overly complex and difficult to use, consider leveraging a wrapper instead in order to simplify the process of storing and retrieving data from the Keychain.

The *Valet* open source library can be used for this purpose: https://github.com/square/Valet.

SECURE CODE

```
// Create a valet instance
VALValet *myValet;
myValet = [[VALValet alloc] initWithIdentifier:@"SecureStorage"
                                 accessibility:VALAccessibilityWhenUnlockedThisDeviceOnly];

// Store a string in the valet
NSString *usernameKey = @"usernameKey";
[myValet setString:@"datatheorem" forKey:usernameKey];

// Retrieve the string
NSString *username = [myValet stringForKey:usernameKey];
```

COMPLIANCES

OWASP Mobile Security
M2 - Insecure Data Storage

MITRE ATT&CK
Obtain Device Cloud Backups (MTC ID: ECO-0, ECO-1)

Logos provided by Clearbit

MEDIA WATCH

Public media articles of other companies who are vulnerable to this issue.

Apple opens up — slightly — on Hong Kong's national security law – TechCrunch

iOS 11 Encrypted Backup Change Reduces Security, Boosts Data Safety

Keybase Bug Might Have Backed up Your Private Encryption Key on Google's Servers