DPAPI security flaw in Windows 10

Passcape Software

1 DPAPI security flaw in Windows 10

1.1 Brief description

Our experts have found a new serious breach in DPAPI security, allowing anyone to decrypt personal data (protected by DPAPI) of the last active user in Windows 10.

1.2 What OS are affected?

This vulnerability affects Windows 10, starting with 1709 Fall Creators Update, as well as Microsoft accounts in previous versions of Windows 10 so far the system volume encryption is activated.

1.3 What is DPAPI?

Data Protection Application Programming Interface (DPAPI) is aimed to perform safe encryption of user's personal data, encryption keys, system-critical data, as well as other sensitive information. It is a primary protection subsystem in all Windows Operating System since Windows 2000. DPAPI has become popular primarily because it is easy to use, as it consists of only two functions to encrypt or decrypting sensitive data: CryptProtectData and CryptUnprotectData. This might sound simple but the internal logic of the CryptProtectData/CryptUnprotectData is quite complex. You can read more about how the DPAPI works in this article.

1.4 Previous DPAPI vulnerabilities

DPAPI was created with many aspects of security in mind and can definitely be considered as one of the best data protection systems, being quite illustrative examples of how a well-designed product can serve for many years. However, the very first implementation had serious troubles. The problem was due to the fact that the primary encryption key in DPAPI v1 was based on NTLM hash of the user's password. This meant that it was just enough to get access to the NTLM hash (that was stored in the SAM registry) to decrypt all passwords and data that were protected by DPAPI. Fortunately, Microsoft promptly found the flaw in logic and rolled out quickly the second DPAPI revision, that's up and running correctly until now.

The new vulnerability is something similar to the first problem, except the following items:

- The new DPAPI issue affects only the last active user of the system
- It does not apply to domain accounts

- Unlike the first implementation, the new vulnerability is not a developer mistake, but a forced compromise of security and usability, so to speak.

1.5 What is Automatic Restart Sign-On (ARSO)?

Starting with Windows 8, it is now possible to launch lock screen applications. That is, applications that start, work and display notifications while the user's session is locked. For example, calendar appointment, notifications, emails, messages, etc. However, during the automatic reboot after an upgrade, these applications would cease to work because they need an active user session. There is an obvious security conflict that has been resolved by Microsoft in a rather original way.

Right before the system initiates an automatic reboot, the current user credentials are stored in a special LSA secret. After rebooting, these credentials are used to automatically log the user in and create an active session, but the interactive part will not be available to the user until he enters the password, PIN, etc. Thus, the last session of the user will be automatically restored and lock screen applications will work. This is how the Automatic Restart Sign-On system works in briefs.

1.6 What is Trusted Boot Auto-Logon (TBAL)

In Windows 10, ARSO uses the Trusted Boot Auto-Logon (TBAL) mechanism. The autologon is a Windows built-in feature that allows to log users on automatically instead of waiting for them to enter their names and passwords. The autologon is activated through the registry where you will need to put in the user's cleartext password. During startup, the system checks the option and if it's set, reads the plaintext password and uses it to perform the logon.

In Windows 10 the autologon was extended with TBAL mechanism. The TBAL is a kind of symbiosis of a common autologon and ARSO features. But it has a number of differences though:

- TBAL supports both regular and Microsoft accounts
- No plaintext passwords are stored by TBAL
- TBAL seems to be always on, not only by request. Although in first versions of Windows 10, system activated TBAL only after the full disk encryption had been enabled.

So how does the TBAL work?

1.7 How does the TBAL work?

Before shutting down, the LSA process saves to the LSA secret **DefaulPassword** a special textual value **_TBAL_{68EDDCF5-0AEB-4C28-A770-AF5302ECA3C9}**, a sign that this is not a common autologon password but a TBAL token instead. Then, depending on the type of the active user account, another LSA secret is created. If this is an offline account, the system stores the user name, NTLM, SHA1 password hashes along with some other private information to the LSA secret named **M$_MSV1_0_TBAL_PRIMARY_{22BE8E5B-58B3-4A87-BA71-41B0ECF3A9EA**. If this is a Microsoft account, then either **M$_CLOUDAP_TBAL_{8283D8D4-55B6-466F-B7D7-17A1352D9CAB}_<UID>** (Windows 1607 and earlier) or **M$_CLOUDAP_TBAL_{4416F0BD-3A59-4590-9579-DA6E08AF19B3}_<UID>** (Windows 1703 and later) secret is created, where **<UID>** is the SHA256 hash of the unique user ID. Unlike the secret for the offline account, this one contains only the 96-byte encryption key needed to derive the DPAPI primary key.

After the PC is rebooted, the system identifies the TBAL token and decrypts the DPAPI primary key using either a SHA1 hash of the user (for an offline account) or a 96-byte key if it is a Microsoft account. Then both LSA token and LSA key secrets are removed.

The TBAL token is not written if the system is hibernating or signing out a user, but only upon reboot or shutdown.

1.8 What is the cause of the problem?

The problem for a user is that after the system is shut down, anyone who has physical access to the PC can use the stored TBAL secret to decrypt the DPAPI primary key and, as a consequence, all the user's data that is encrypted using DPAPI. It is obvious that the cause of the vulnerability is not the misbehaved logic, but the conceptual approach of Microsoft to the Windows security, unlike one that was found in the first DPAPI implementation or unlike in previous errors in password-free login implementation. However, this seems to be a global trend of recent years.

1.9 What data is at risk?

- Network passwords saved by popular Internet browsers: Google Chrome, Internet Explorer, Microsoft Edge, Opera, etc.
- Passwords of email clients: Microsoft Office Outlook, Windows Mail.
- Passwords to shared folders and resources
- Passwords, keys and other private data stored in Windows Vault
- Remote Desktop passwords
- EFS private keys and thus access to EFS encrypted files
- Encryption keys in S-MIME mail

- Users' certificates
- Network passwords stored in Credential Manager
- Any personal data protected using CryptProtectData API in any application, such as Skype, Windows Rights Management Services, Windows Media, MSN messenger, Google Talk, etc.

1.10 PoC

This video demonstrates how easy it is to access the personal data of the last active user without knowing his/her login password in Windows 10. Even though it is assumed that no one can do this without knowing the owner's logon password, the program uses TBAL secret to decrypt Facebook credentials stored in Windows Vault and protected by DPAPI.

1.11 Conclusion

As we warned in our previous article, the next versions of Windows will become less and less focused on ensuring the safety for an end-user. Those users for whom it is important to provide the maximum level of protection, it is recommended to set up an offline account along with Windows 10 up to version 1709 with the SYSKEY startup password or full disk encryption set on.