DPAPI exploitation during pentest and password cracking
whoami /groups

- Jean-Christophe Delaunay - @Fist0urs
- Jiss/Fist0urs on IRC
- Synacktiv – www.synacktiv.ninja

- Microsoft Windows Active Directory (kerberom)
- Passcracking - User and contributor to John The Ripper and hashcat (krb5tgs, axcrypt, keepass, dpapimk, etc.)
Roadmap

- What is DPAPI?
- For real, what is DPAPI?
- DPAPI during pentest
- What’s next?
- Questions
What is DPAPI – a bit of history

- Data Protection Application Programming Interface
- Helps protect secrets (passwords, certificates, etc.)
- Exists since *Windows 2000*
- Evolved a lot but core is globally the same
- Transparent for the end-users
What is DPAPI – wtfbbq?

- Cryptography based on user’s password (not exactly in fact)
- Easy to implement for developers:
  - `CryptProtectData`
  - `CryptUnprotectData`
- Widely used:
  - Credential Manager, Windows Vault, IE, Wi-Fi, Certificats, VPN, etc.
  - Google Chrome, Google Talk, Skype, Dropbox, iCloud, Safari, etc.
DPAPI Internals

- DPAPI is:
  - Transparent for the end-users
  - Easy to use for developers
  - … Hard when you want to really understand the internals
DPAPI Internals – developers view

Application

encrypted Blob

Data to protect

CrypProtectData

DPAPI
DPAPI Internals – reverser view
DPAPI Internals – developers view

```c
BOOL WINAPI CryptProtectData(
    _In_     DATA_BLOB *pDataIn,
    _In_opt_ LPCWSTR szDataDescr,
    _In_opt_ DATA_BLOB *pOptionalEntropy,
    _Reserved_ PVOID pvReserved,
    _In_opt_ CRYPTPROTECT_PROMPTSTRUCT *pPromptStruct,
    _In_     DWORD dwFlags,
    _Out_    DATA_BLOB *pDataOut
);
```
DPAPI Internals – crypto
DPAPI Internals – crypto

- Secret based on user’s password… is it sufficient?
  - what about password changing?
  - what about *Rainbow Tables* attacks?
DPAPI Internals – crypto

- Secret based on user’s password… is it sufficient?
  - what about password changing?
  - what about *Rainbow Tables* attacks?

- … but this is not sufficient, *master keys* are used. These masterkeys are stored in *blobs*, each containing:
  - a GUID
  - a *salt*
  - *master key* structure (containing *master keys*)
DPAPI Internals – DPAPI Blob

DWORD dwVersion
[ ... ]
GUID guidMasterKey
ALG_ID algCrypt
DWORD dwCryptAlgLen
BYTE pSalt[dwSaltLen]
BYTE pHmac[dwHmacKeyLen]
ALG_ID algHash
DWORD dwHashAlgLen
[ ... ]
BYTE pData[dwDataLen]
BYTE pSign[dwSignLen]
DPAPI Internals – *master keys*

DWORD dwVersion

[ ... ]

GUID `guidMasterKey`

ALG_ID `algCrypt`

DWORD `dwCryptAlgLen`

BYTE `pSalt[dwSaltLen]`

BYTE `pHmac[dwHmacKeyLen]`

ALG_ID `algHash`

DWORD `dwHashAlgLen`

[ ... ]

BYTE `pData[dwDataLen]`

BYTE `pSign[dwSignLen]`
DPAPI Internals – *master key* header

<table>
<thead>
<tr>
<th><em>master key structure header</em></th>
<th>dwVersion;</th>
</tr>
</thead>
<tbody>
<tr>
<td>user master key</td>
<td>[ ... ]</td>
</tr>
<tr>
<td>CREDHIST, or something else...</td>
<td>[ ... ]</td>
</tr>
<tr>
<td>local encryption key</td>
<td>szGuid[0x24];</td>
</tr>
<tr>
<td>domain backup key</td>
<td>dwUserKeySize;</td>
</tr>
<tr>
<td></td>
<td>dwLocalEncKeySize;</td>
</tr>
<tr>
<td></td>
<td>dwLocalKeySize;</td>
</tr>
<tr>
<td></td>
<td>dwDomainKeySize;</td>
</tr>
</tbody>
</table>
DPAPI Internals – WTH is he talking about?...

Application → User logon context → User hash → Key

PBKDF2: (hash + SID + salt)

DPAPI Blob → Decrypted masterkey + GUID

GUID → Protected secret

URL: https://facebook.com
Login: Fist0urs
Password: Univershell_facebook
DPAPI Internals – can I attack it?

<table>
<thead>
<tr>
<th>OS</th>
<th>Ciphering algo</th>
<th>Hashing algo</th>
<th>PBKDF2 iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows 2000</td>
<td>RC4</td>
<td>SHA1</td>
<td>1</td>
</tr>
<tr>
<td>Windows XP</td>
<td>3DES</td>
<td>SHA1</td>
<td>4000</td>
</tr>
<tr>
<td>Windows Vista</td>
<td>3DES</td>
<td>SHA1</td>
<td>24000</td>
</tr>
<tr>
<td>Windows 7</td>
<td>AES256</td>
<td>SHA512</td>
<td>5600</td>
</tr>
<tr>
<td>Windows 10</td>
<td>AES256</td>
<td>SHA512</td>
<td>8000</td>
</tr>
</tbody>
</table>
DPAPI Internals – CREDHIST

- Is used to decrypt master keys protected by older passwords
- Stores all previous passwords’ hashes
- An old hash is protected by the first most recent one
- Stores hashes in NTLM and SHA1 formats
DPAPI Internals – CREDHIST

1

\[\text{NT/SHA1 hash} \rightarrow \text{encryption} \rightarrow \text{NT/SHA1 hash} \rightarrow \cdots \rightarrow \text{NT/SHA1 hash} \rightarrow \text{PBKDF2} \rightarrow \text{password} \]
DPAPI Internals – what’s next…?

- *master keys* backup?
- Entropy?
- DPAPI system?
- SHA1 and NTLM?
- What about domain and local contexts?
DPAPI Internals – stored...?

- In the user’s profile (%APPDATA%/Roaming/Microsoft)
  - Protect/CREDHIST
  - Protect/SID
  - Protect/SID/Preferred
  - Credentials
  - Vault
  - etc.

- In the registry
- In system32
- etc.
DPAPI – pentest

- 2 possibilities:
  - I can execute some code on the remote host
  - I can’t...
DPAPI – existing tools

- Passcape: shareware + Windows only [1]
- impact: does not decrypt DPAPI protected secrets directly [2]
- mimikatz: extracts secrets online and offline but Windows only [3]
- dpapick: extracts secrets offline! First tool published to manage DPAPI offline, incredible work! [4]
- dpapilab: an extension of dpapick [5]
DPAPI – what can I do? I can execute commands

- I am in user’s (not admin!) authentication context but do not have his password:
  - Use Windows API to extract some DPAPI protected secrets, using implicit authentication (*mimikatz, CredMan.ps1*, etc.)
  - But I would like to have his session password...
DPAPI – what can I do? I can execute commands

- But I would like to have his session password...
- Wait, you told us that secrets are protected by user’s password?...
- ...and master keys are also protected by user’s password?
- ...
- Profit! (format merged in John the Ripper yesterday \o/) [6]
DPAPI – what can I do? I can execute commands

$ python DPAPImk2john.py -h

optional arguments:
  -h, --help                           show this help message and exit
  -S SID, --sid SID                    SID of account owning the masterkey file.
  -mk MASTERKEY, --masterkey MASTERKEY masterkey file (usually in %APPDATA\%\Protect\<SID>).
  -d, --debug                         context of user account. Only 'domain' and 'local' are possible.
  -c CONTEXT, --context CONTEXT       'Preferred' file containing GUID of masterkey file in use (usually in %APPDATA\%\Protect\<SID>). Cannot be used with any other command.
  -P PREFERRED, --preferred PREFERRED password to decrypt masterkey file.
  --password PASSWORD

SYNAACKTIV
DIGITAL SECURITY
DPAPI – what can I do? I can execute commands

```
Fist0urs@mongodabest:~/univershell$ python DPAPImk2john.py -P Preferred 1b4ac82b-1a40-456e-83bb-ca5e1d91024c
```
DPAPI – what can I do? I can execute commands

```bash
Fist0urs@mongodbest:~/univershell$ python DPAPImk2john.py
--sid="S-15-21-478900483-410193244-460175230-1818"
--masterkey= "1b4ac82b-1a40-456e-83bb-ca5e1d91024c"
--context="local"

$DPAPImk$1*1*S-15-21-478900483-410193244-460175230-1818*des3*sha1*24000*2c227152554a45e37ebef7d244c8bc85*208*
6d7b48964c5a451ee267c46abf31a5d67980f4b738629d65cb65534daa
d9bd252eb25af55dc08d514b2385cf9bf3575ff8954b764b4175467d76
ee5bbdb52dd29e1aa012129486d7de38e3a7a1dc059fe4a0aab2a5c16c
93f6d592b9616333ebbce5016036d58aad
```
DPAPI – what can I do? I can execute commands

Fist0urs@mongodabest:$ john
univershell.dump --wordlist=dpapi_extracted.dic
--rules=custom.rule

Using default input encoding: UTF-8
Loaded 1 password hash (DPAPImk, DPAPI masterkey file v1 and v2 [SHA1/MD4 PBKDF2-(SHA1/SHA512)-DPAPI-variant 3DES/AES256 256/256 AVX2 8x])
Will run 8 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
Univershell_synacktiv (?)
1g 0:00:00:00 DONE (2017-04-26 12:07) 4.761g/s 14.28p/s 14.28c/s 14.28C/s ..Univershell_synacktiv
Use the "--show" option to display all of the cracked passwords reliably
Session completed
DPAPI – what can I do? I can not execute commands

- I still can get the masterkey files and retrieve user’s password, but no dpapi stuff
- ...for the moment!
DPAPI – meet *dpapeace*

- Based on work done on *dpapick* and *dpapilab* + its Core
- Recoded and completed what *dpapick* et *dpapilab* do
- Plugins handling
- parser/writer handling (XML only at the moment)
- Still a POC for the moment...
<?xml version="1.0"?>
<dpapi>
  <computer name="Fist0urs-PC" ip="192.168.0.1">
    <hives>
      <system>/home/Fist0urs/DPAPI/DATA/sys/sys</system>
      <security>/home/Fist0urs/DPAPI/DATA/sys/sec</security>
    </hives>
    <sysmasterkey>/home/Fist0urs/DPAPI/DATA/sys/S-1-5-18/User/</sysmasterkey>
    <wifi>/home/Fist0urs/DPAPI/DATA/Wifi/Wlansvc/Profiles/Interfaces/{747AXXXX-XXXX-XXXX-XXXX-XXXX81530EE7}</wifi>
    <account name="Fist0urs" sid="S-1-5-21-478900483-410193244-460175230-1818" domain="WORKGROUP">
      <masterkey>/home/Fist0urs/DPAPI/DATA/Protect/S-1-5-21-478900483-410193244-460175230-1818</masterkey>
    </account>
    <credhist>/home/Fist0urs/DPAPI/DATA/Protect/CREDHIST</credhist>
    <credentials>
      <password>Univershell_synacktiv</password>
      <context>local</context>
      <hash>**</hash>
    </credentials>
    <worker name="chrome">
      <target>/home/Fist0urs/DPAPI/DATA/Chrome/Login Data</target>
    </worker>
    <worker name="credman">
      <target>/home/Fist0urs/DPAPI/DATA/Credentials</target>
    </worker>
    <worker name="winvault">
      [...]
    </worker>
  </computer>
</dpapi>
<?xml version="1.0"?>
<dpapi>
  <computer ip="192.168.0.1" name="Fist0urs-PC">
    <account domain="WORKGROUP" name="Fist0urs" sid="S-1-5-21-478900483-410193244-460175230-1818">
      <credentials type="chrome">
        <url name="http://crackmes.de/">
          <username>Fist0urs</username>
          <password>****</password>
        </url>
        <url name="https://websec.fr/login">
          <username>Fist0urs</username>
          <password>****</password>
        </url>
      </credentials>
      <credentials type="credman">
        <cred persist="Entreprise" type="Domain password">
          <target>Domain:target=trolololol.fr</target>
          <username>mwa</username>
          <password>**</password>
          <last_modified>2016-09-17T20:33:32+00:00</last_modified>
        </cred>
        <cred persist="Entreprise" type="Domain password">
          <target>Domain:Fist0urs@timmy.com</target>
          <username>Fist0urs</username>
          <password>****</password>
          <last_modified>2016-09-18T17:43:20+00:00</last_modified>
        </cred>
    </account>
</computer>
[...]

34 / 39
DPAPI – pentest conclusions

- It is really useful during pentests:
  - Retrieve many secrets protected by user’s password
  - Possibly retrieve user’s password (useful when phishing or exploiting a context-based vulnerability)
  - Also an alternative to MSCashvX (if admin), in case a workstation is harden (0 or 1 credential cached) as masterkeys are imported in his roaming profil when one connects interactively on a workstation
  - Much more stealth as it only requires to copy some files from the filesystem
  - Difficult to spot :)

---

35 / 39
DPAPI – pentest

Fist0urs@mongodabest:/univershell$ john --format=mscash2 --test && john --format=dpapidmk --test
Will run 8 OpenMP threads
Benchmarking: mscash2, MS Cache Hash 2 (DCC2) [PBKDF2-SHA1 256/256 AVX2 8x]... (8xOMP) DONE
Warning: "Many salts" test limited: 19/256
Many salts: 9228 c/s real, 1225 c/s virtual
Only one salt: 8447 c/s real, 1152 c/s virtual

Will run 8 OpenMP threads
Benchmarking: DPAPIDmk, DPAPI masterkey file v1 and v2 [SHA1/MD4 PBKDF2-(SHA1/SHA512)-DPAPI-variant 3DES/AES256 256/256 AVX2 8x]... (8xOMP) DONE
Speed for cost 1 (iteration count) of 24000
Raw: 2115 c/s real, 256 c/s virtual

Not that bad regarding the iterations count!
DPAPI – future work

1) Implement the algorithm in John the Ripper
2) Implement the algorithm in hashcat
3) Continue development of dpapeace (in particular Windows implicit authentication)
4) Publish dpapeace once everything is clean
5) More :)
ANY QUESTIONS?

Thank you for your attention!

SYNACKTIV
Bibliography

[1] https://www.passcape.com/