# Simplifier l'authentification avec Kerberos

## Du mono-poste à la PME

Matthieu CERDA

Normation

Mardi 10 Juillet 2012

## Qui suis-je ?

Matthieu CERDA

Administrateur réseaux et systèmes chez Normation

## Tu fais quoi dans la vie ?

- Gère l'infrastructure informatique chez Normation
- Travaille sur l'outil de gestion de configuration Rudder
- Missions d'expertise sur LDAP, LSC, CFEngine ...
- Passionné de systèmes libres, et de sécurité

## Kerberos

Kerberos est un protocole normalisé d'authentification réseau, utilisant de la cryptographie forte

## But de cette présentation

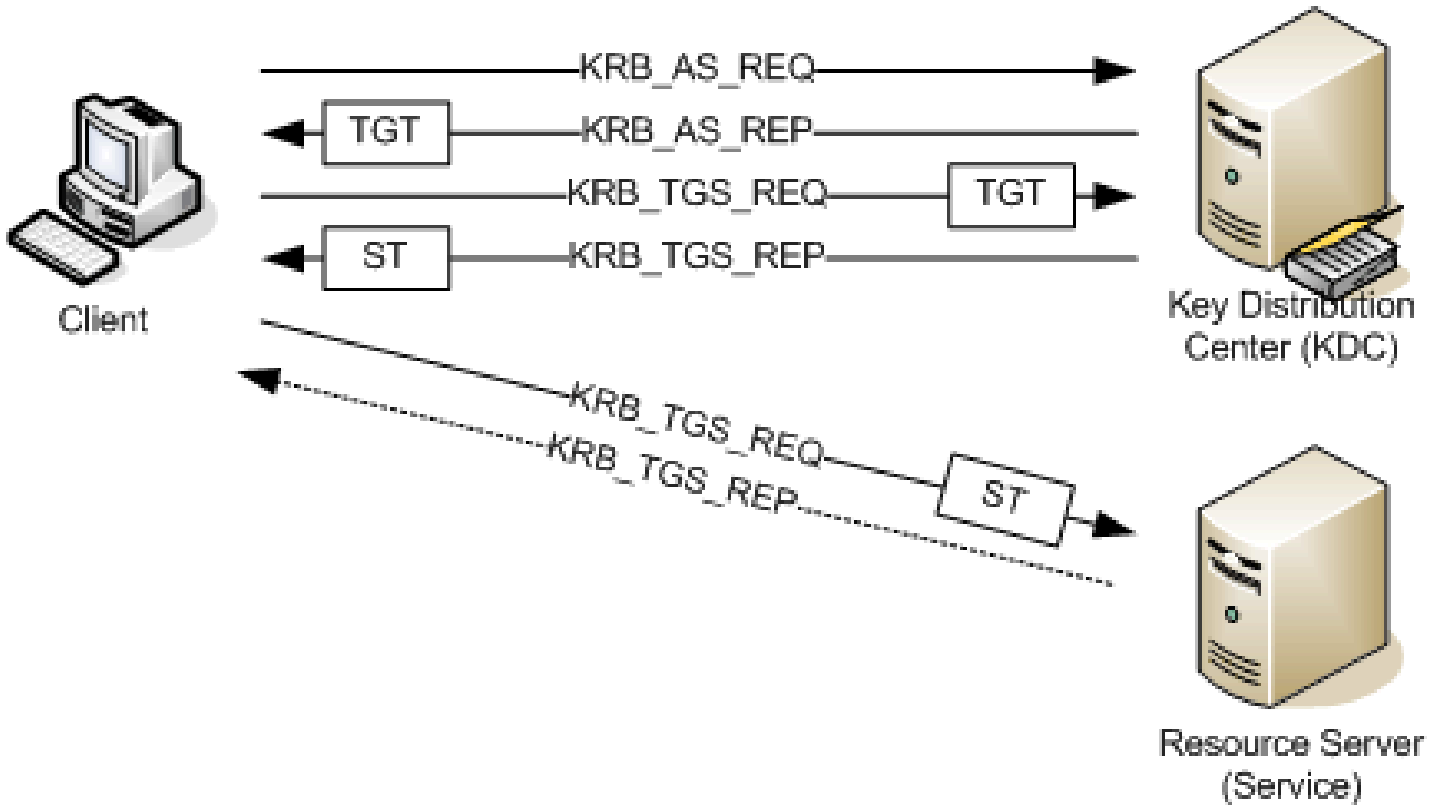
Cette présentation a pour but de présenter Kerberos dans un contexte d'utilisation quotidienne ...

... et de démontrer que c'est un outil moins compliqué que ce qu'on pense souvent !

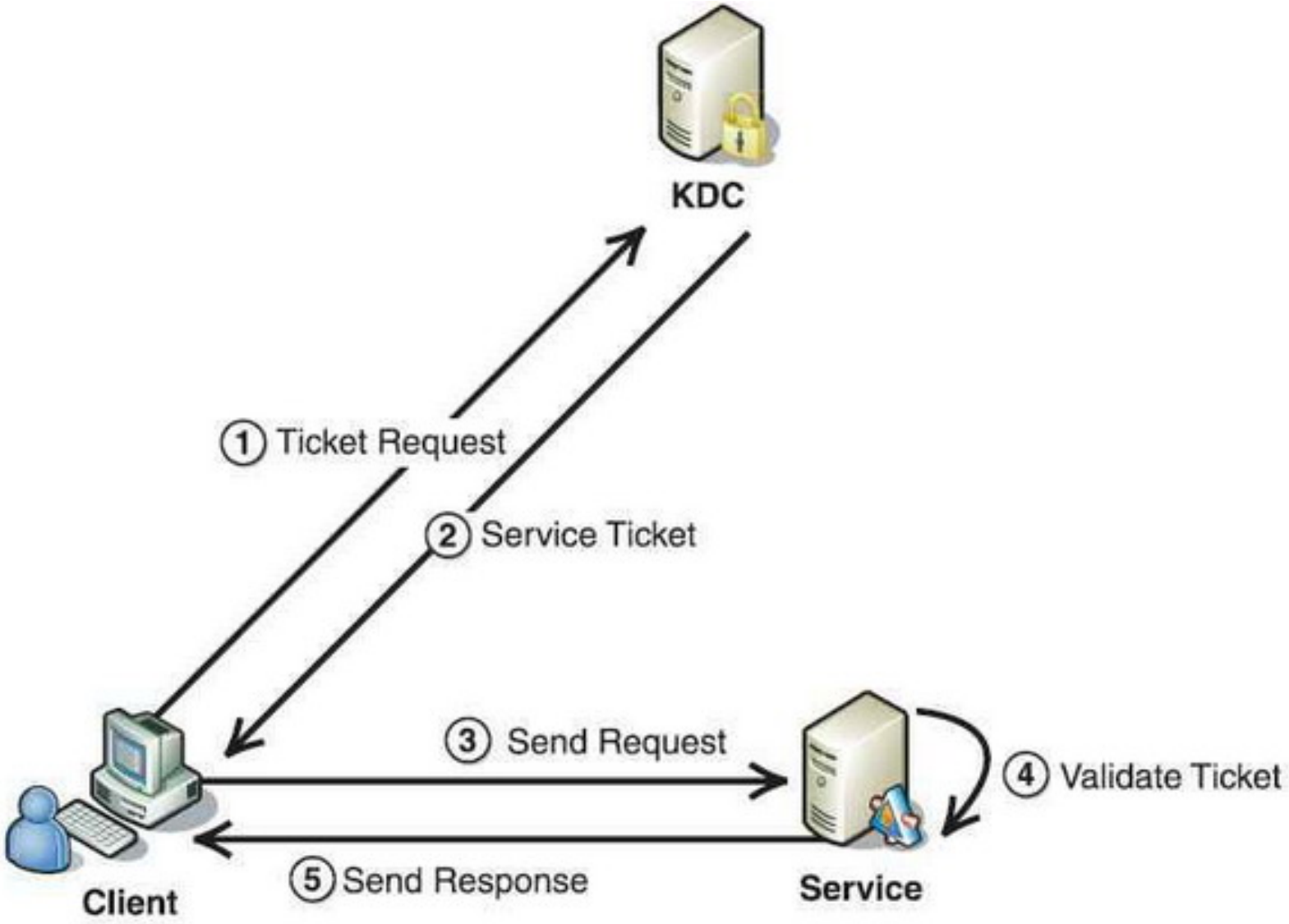
# Kerberos

- Protocole d'authentification réseau
- Très largement déployé, mais peu connu
- Chaque élément est séparé et communique via réseau
- Très strict avec les noms de domaine qualifiés et les horloges
- Sécurisé via des algorithmes symmétriques comme AES ou Blowfish/Twofish
- Peut être utilisé nativement avec beaucoup d'outils (NFS, LDAP, SMTP, SSH, HTTP...)

# Kerberos


KRB_AS_REQ
TGT
KRB_AS_REP
KRB_TGS_REQ
TGT
ST
KRB_TGS_REP
Client
Key Distribution Center (KDC)
KRB_TGS_REQ
KRB_TGS_REP
ST
Resource Server (Service)

# Kerberos


KDC
1 Ticket Request
2 Service Ticket
3 Send Request
4 Validate Ticket
5 Send Response
Client
Service

# Kerberos V5

L'implémentation actuelle la plus utilisée de Kerberos est sa version 5 :

## Avantages

1. Très largement déployée
2. Préinstallée sur beaucoup de systèmes d'exploitation
   - GNU/Linux
   - *BSD
   - Windows !
3. Bien intégrée au système grâce a la GSSAPI
4. Facile ( !) a installer

## Inconvénients

1. L'intégration a la distribution est parfois peu documentée
2. Kerberos fait peur ! ! !

# Kerberos V5

L'implémentation actuelle la plus utilisée de Kerberos est sa version 5 :

## Avantages

1. Très largement déployée
2. Préinstallée sur beaucoup de systèmes d'exploitation
   - GNU/Linux
   - *BSD
   - Windows !
3. Bien intégrée au système grâce a la GSSAPI
4. Facile ( !) a installer

## Inconvénients

1. L'intégration a la distribution est parfois peu documentée
2. Kerberos fait peur ! ! !

# Kerberos V5

L'implémentation actuelle la plus utilisée de Kerberos est sa version 5 :

## Avantages

1. Très largement déployée
2. Préinstallée sur beaucoup de systèmes d'exploitation
   - GNU/Linux
   - *BSD
   - Windows !
3. Bien intégrée au système grâce a la GSSAPI
4. Facile ( !) a installer

## Inconvénients

1. L'intégration a la distribution est parfois peu documentée
2. Kerberos fait peur ! ! !

# Kerberos V5

L'implémentation actuelle la plus utilisée de Kerberos est sa version 5 :

## Avantages

1. Très largement déployée
2. Préinstallée sur beaucoup de systèmes d'exploitation
   - GNU/Linux
   - *BSD
   - Windows !
3. Bien intégrée au système grâce a la GSSAPI
4. Facile ( !) a installer

## Inconvénients

1. L'intégration a la distribution est parfois peu documentée
2. Kerberos fait peur ! ! !

# Deux implémentations majeures

Il existe deux implémentations majeures de Kerberos dans les systèmes libres :

MIT

1. L'implémentation "historique"
2. Longtemps restreinte a l'export hors des U.S.A.
3. Bien connue, robuste
4. Bien documentée
5. Beaucoup de systèmes utilisent MIT comme libkrb5

# Deux implémentations majeures

Il existe deux implémentations majeures de Kerberos dans les systèmes libres :

## MIT

1. L'implémentation "historique"
2. Longtemps restreinte a l'export hors des U.S.A.
3. Bien connue, robuste
4. Bien documentée
5. Beaucoup de systèmes utilisent MIT comme libkrb5

## Heimdal

1. Une version concurrente issue des restrictions d'export de MIT
2. Réputée pour avoir une base de code plus "propre"
3. Les outils d'administration supportent readline
4. Plus "facile" a utiliser pour un utilisateur novice
5. Préinstallée par défaut sous OpenBSD

Par la suite, j'utiliserai heimdal, mais c'est un choix purement arbitraire, sachant que les deux versions sont relativement proches au niveau de la syntaxe.

# Installation sous Debian

- C'est sous Debian que l'installation est la plus aisée !
- Laisser la magie d'APT opérer.

```
root@box-# aptitude install heimdal-kdc heimdal-clients
```

- Répondre aux questions qui seront posées. Dans notre cas, considérons que le domaine est ZEA.ZEN et que le KDC est pinkie.zea.zen

# Installation sous Debian

- C'est sous Debian que l'installation est la plus aisée !
- Laisser la magie d'APT opérer.

```
root@box-# aptitude install heimdal-kdc heimdal-clients
```

- Répondre aux questions qui seront posées. Dans notre cas, considérons que le domaine est ZEA.ZEN et que le KDC est pinkie.zea.zen


# Installation sous RHEL/CentOS/Fedora

- Sous les OS RPM en général, il faut en revance mettre la main a la pâte car il n'y a pas de magie a la debconfig.
- Cette fois, seule MIT est disponible. L'installer via votre package manager préféré :

```
root@box-# yum install krb5-workstation
```

- Cette fois ci, il faut faire la configuration a la main. Je ne peux que conseiller soit de récupérer le fichier /etc/krb5.conf sur le site de MIT Kerberos ou de compléter le fichier existant en s'aidant de la documentation.

# Installation sous OpenBSD

- "Ha mais ça doit être méga compliqué, si c'est sous OpenBSD ! ! !"
- Mais non. C'est simple et en plus très bien documenté.
- Une fois n'est pas coutume, c'est presque aussi simple que sur les OS RPM :

```
root@box-# echo krb5_master_kdc=YES >> rc.conf.local
root@box-# mkdir /var/heimdal
root@box-# kstash --random-key
root@box-# kadmin -l
kadmin-# init ZEA.ZEN
```

- On remplace ensuite les MY.REALM et l'adresse du kdc de MY.REALM dans le fichier /etc/kerberosV/krb5.conf et tout est prêt.

# Utilisation générale de Kerberos

- kadmin -l : En root, permet d'administrer la base Kerberos en local
  - init : Initialise un royaume Kerberos
  - add : Ajoute un principal (avec –random-password pour un principal de service)
  - ext_keytab : Exporte un keytab (–keytab pour choisir la destination du keytab)
- kinit : Lance un AS-REQ et ainsi récupère un TGT.
- klist : Permet de voir les TGT stockés actuellement
- Keytabs : Copie locale d'un trousseau de principaux de service pour pouvoir vérifier les tickets de services envoyés par le client

# Outils utilisés

1. libpam_krb5 : Permet d'utiliser Kerberos avec PAM (kinit automatique au login, modification de mots de passe...)
2. sshd : L'antique démon Secure Shell libre
3. slapd : Le démon LDAP du projet OpenLDAP
4. httpd : Le démon HTTP du projet Apache

# Kerbérisation de SSHd

- Exporter un principal host/(fqdn de la machine)@ZEA.ZEN

# Kerbérisation de SSHd

- Exporter un principal host/(fqdn de la machine)@ZEA.ZEN
- Editer deux paramètres dans /etc/ssh/sshd_config :
  - KerberosAuthentication : Valide le mot de passe auprès du KDC
  - GSSAPIAuthentication : Active l'authentification via GSSAPI
- On oublie pas de relancer SSHd. Pas d'inquiétude, ça ne déconnecte pas les sessions actives.

```
client@box-# ssh -o "GSSAPIAuthentication off" pinkie.zea.zen
client's password: <= DO NOT WANT.
client@box-# ssh -o "GSSAPIAuthentication on" pinkie.zea.zen
client@pinkie-#
```

# Kerbérisation de Apache HTTPd

- Exporter un principal HTTP/(fqdn de la machine)@ZEA.ZEN vers /etc/apache2/
- Insérer dans la configuration du vhost :

```
<Location /secured>
    AuthType Kerberos
    KrbMethodNegotiate on
    KrbMethodK5Passwd on
    AuthName "Kerberos Login"
    KrbAuthRealms ZEA.ZEN
    Krb5Keytab /var/www/conf/http.keytab
    require valid-user
</Location>
```

# Kerbérisation de NFS

- Exporter un principal nfs/(fqdn de la machine)@ZEA.ZEN dans la keytab par défaut
- Configurer rpc.idmapd et rpc.gssd, et choisir son niveau de chiffrement avec mount :
  - sec=krb5 : Authentification uniquement
  - sec=krb5i : Contrôle d'intégrité
  - sec=krb5p : Chiffrement des flux

# Kerbérisation de Samba

- Exporter un principal cifs/(fqdn de la machine)@ZEA.ZEN dans la keytab par défaut
- Configurer smb.conf

# Kerberos en entreprise ?

- Vous l'utilisez peut etre déja sans le savoir, si vous etes dans un domaine Active Directory
- Kerberos est tout a fait a meme de se placer en tant que couche de sécurité principale ou complémentaire dans beaucoup de cas
  - Logins via LDAP
  - Sécurisation des transferts de données
  - Plus de besoin de manager des clés privés SSH

# Single Sign On - SSO

- Principal avantage de Kerberos en entreprise
- Si on configure correctement Kerberos, l'utilisateur ne se loggue qu'une fois et accède a ses ressources sans etre dérangé.
  - Sécurité : L'utilisateur est moins tenté de laisser ses mots de passe trainer ...
  - Confort : L'utilisateur n'a pas a entrer tout le temps son mot de passe
  - Sureté : Les algorithmes de chiffrement utilisés par Kerberos sont dans leur majorité réputés surs.
  - Unicité : Les comptes sont gérés a un seul endroit, sans avoir a changer des fichiers partout

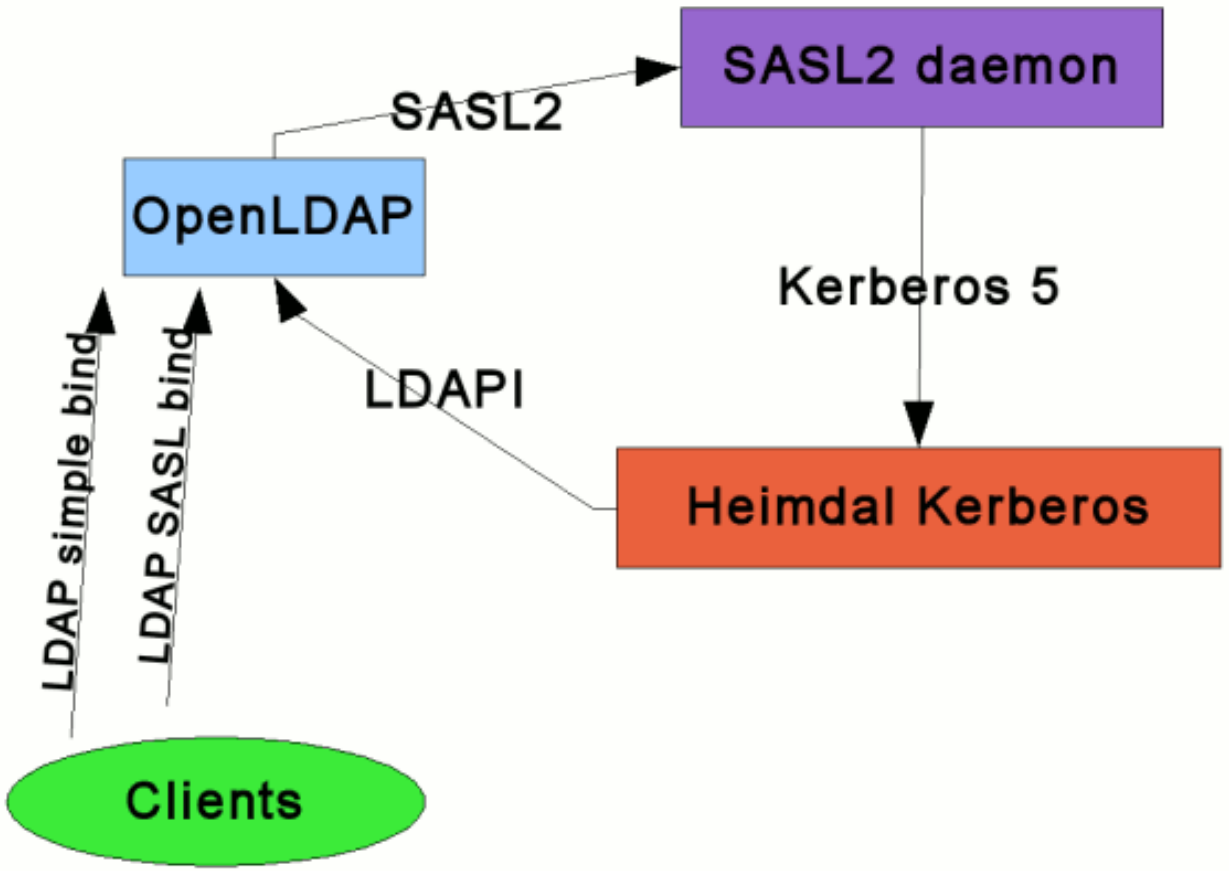
# LDAP

- Le meilleur ami de Kerberos dans un contexte de SSO
- LDAP assure la définition des informations de l'utilisateur, Kerberos la couche de sécurité

# SASL

- Protocole générique de transport de données authentifiées
- SASL est utilisé par de nombreux autres protocoles comme SMTP, IMAP, POP, LDAP, ...
- Dans notre cas, SASL assure la sécurité du transport des données LDAP et la liaison Kerberos - LDAP

# Parceque les mots, ça va bien 5 minutes ...




# Résumé

1. LDAP assure la définition des comptes utilisateurs
2. Kerberos assure l'authentification persistante (SSO)
3. SASL sert de mécanisme d'authentification entre les deux via la GSSAPI

# Fusion de Kerberos et LDAP

1. On obtient un référentiel d'authentification unique
2. Demande un peu plus d'efforts
   - Des ACLs OpenLDAP a créer
   - Des mappings a faire entre SASL et LDAP
3. Permet d'obtenir un service d'annuaire complet et sécurisé libre

# Plus loin.. ?

1. Active Directory n'est pas forcément ton ennemi Bon, si peut être un peu ...
2. Frameworks de SSO tout faits comme LemonLDAP : :NG (`http://lemonldap-ng.org`)
3. La GSSAPI est assez simple a utiliser, pourquoi ne pas l'intégrer a votre programme ?

# Références


1. Le tout puissant et saint manuel ! (man heimdal)
2. Documentation Debian et OpenBSD
3. GLMF Num. 143 - Article de Guillaume Rousse (Coucou Guillaume !)


Avez-vous des questions ?

Merci pour votre attention !