Kerberos / Active Directory / CAS

La doc de CAS est assez succincte et on ne sait pas toujours quoi mettre. Ici à Aix-Marseille Université, nous avons pas mal galéré pour ajouter un load balancer haproxy avec adresse IP virtuelle.

Dans notre cas, notre domaine est salsa.univ-amu.fr, et notre cas de test cas-test.univ-amu.fr. Bref, que faut il mettre et générer. On doit générer un keytab correspondant à un service en utilisant un compte windows spécialement créé pour le besoin. Ce compte windows est lié au keytab généré et on ne doit pas l'utiliser pour autre chose.

On génère donc un keytab pour un service. Et attention, le service ne DOIT PAS ÊTRE UN CNAME ! Si besoin on transforme le CNAME en A. Donc on a mis un deuxième A sur la VIP du load-balancer. Ce keytab doit être pointé dans le krb5.conf. On ne parle que de Firefox, car les autres navigateurs posent problème. S'ils ne sont pas dans le domaine AD et ne peuvent pas faire de kerberos, il basculent en NTML avec une interface pourrie style AUTH/BASIC.

Pour que Firefox active le mode Kerberos, il faut utiliser about:config et mettre ceci qui peut être aussi posé par des policies windows :

network.negotiate-auth.trusted-uris = univ-amu.fr

krb5.conf

```
[libdefaults]
        default_realm = SALSA.UNIV-AMU.FR
        default_keytab_name = /etc/cas/config/kerberos/cas-test.keytab
        dns_lookup_realm = true
        dns_lookup_kdc = true
        default_tkt_enctypes = rc4-hmac
        default_tgs_enctypes = rc4-hmac

[realms]
        SALSA.UNIV-AMU.FR = {
                kdc = xxx.salsa.univ-amu.fr:88
                kdc = yyy.salsa.univ-amu.fr:88
        }
[domain_realm]
        .salsa.univ-amu.fr = SALSA.UNIV-AMU.FR
        salsa.univ-amu.fr = SALSA.UNIV-AMU.FR
```

klist -k sur le serveur CAS

```
Keytab name: FILE:/etc/cas/config/kerberos/cas-test.keytab
KVNO Principal
---- --------------------------------------------------------------------------
   3 HTTP/cas-test.univ-amu.fr@SALSA.UNIV-AMU.FR
```

Le serveur réel est dans un autre sous domaine derrière le haproxy et ça ne pose pas de problème.

Problèmes rencontrés qui ont pris pas mal de temps:

1. Le CNAME
2. Les essais avec différents keytab. Attention, le client windows garde en cache des tickets qui peuvent être dans un sale état. Il faut purger les tickets sur le poste avec la commande windows **klist purge**
3. Firefox web developper oublie de présenter 2 échanges HTTP qui ne permettent pas de comprendre comment ça marche. On s'en est aperçu avec Wireshark. Wireshark peut filtrer les contenus kerberos ce qui rend lisible les échanges.
4. Et Firefox doit aussi garder des choses en cache. Un redémarrage de Firefox a aussi résolu un problème.

Tout ça combiné peut faire perdre pas mal de jours.

Le client windows Firefox reçoit une demande d'authentification HTTP/kerberos liée au service CAS. Il demande au KDC un ticket pour ce service et va le renvoyer au serveur qui voit une réponse valide. Pour le serveur, nul besoin de contacter les KDC. Il voit que le ticket retourné est valide.

Sur mon poste windows, klist va me montrer mes tickets Kerberos dont celui lié à CAS

klist sur la machine windows

```
#3>     Client : monuid @ SALSA.UNIV-AMU.FR
        Serveur : HTTP/cas-test.univ-amu.fr @ SALSA.UNIV-AMU.FR
        Type de chiffrement KerbTicket : RSADSI RC4-HMAC(NT)
        Indicateurs de tickets 0x40a10000 -> forwardable renewable pre_authent name_canonicalize
        Heure de démarrage : 6/22/2022 13:59:00 (Local)
        Heure de fin :   6/22/2022 23:43:14 (Local)
        Heure de renouvellement : 6/29/2022 13:43:14 (Local)
        Type de clé de session : RSADSI RC4-HMAC(NT)
        Indicateurs de cache : 0
        KDC appelé : kdc1.salsa.univ-amu.fr
```

Pas besoin de mettre un login.conf dans votre configuration

cas.properties

```
cas.authn.spnego.mixedModeAuthentication=true
#cas.authn.spnego.supportedBrowsers=MSIE,Trident,Firefox,AppleWebKit
cas.authn.spnego.supportedBrowsers=Firefox
cas.authn.spnego.send401OnAuthenticationFailure=false
cas.authn.spnego.ntlmAllowed=false
cas.authn.spnego.principalWithDomainName=false
cas.authn.spnego.name=spnego
cas.authn.spnego.ntlm=false
cas.authn.spnego.order=1
cas.authn.spnego.system.kerberos-conf=file:/etc/krb5.conf
cas.authn.spnego.system.kerberosRealm=SALSA.UNIV-AMU.FR
cas.authn.spnego.properties[0].jcifsServicePrincipal=HTTP/cas-test.univ-amu.fr@SALSA.UNIV-AMU.FR
cas.authn.spnego.properties[0].jcifsDomain=salsa.univ-amu.fr
#cas.authn.spnego.system.kerberosDebug=true
cas.authn.spnego.hostNameClientActionStrategy=hostnameSpnegoClientAction
cas.authn.spnego.ipsToCheckPattern=^(10.*|172.*)$
```

Une référence intéressante sur Kerberos et HTTP : http://remivernier.com/index.php/2018/09/16/exploration-des-entetes-http-www-authenticate/

Mise à jour des algorithmes de chiffrements sur les KDC

RC4-HMAC est considéré comme faible, et une récente (8 novembre 2022) mise à jour des AD (https://support.microsoft.com/help/5021131) peut provoquer des dysfonctionnement avec l'authentification KERBEROS.

C'est un problème que nous avons rencontré à l'AMU en constatant que SPNEGO ne fonctionnait plus et des erreurs dans les logs :

```
Caused by: GSSException: Failure unspecified at GSS-API level (Mechanism level: Encryption type RC4 with HMAC
is not supported/enabled)
    at java.security.jgss/sun.security.jgss.krb5.Krb5Context.acceptSecContext(Krb5Context.java:859)
    at java.security.jgss/sun.security.jgss.GSSContextImpl.acceptSecContext(GSSContextImpl.java:361)
    at java.security.jgss/sun.security.jgss.GSSContextImpl.acceptSecContext(GSSContextImpl.java:303)
    ... 267 more
Caused by: KrbException: Encryption type RC4 with HMAC is not supported/enabled
    at java.security.jgss/sun.security.krb5.EncryptionKey.findKey(EncryptionKey.java:544)
    at java.security.jgss/sun.security.krb5.KrbApReq.authenticate(KrbApReq.java:273)
    at java.security.jgss/sun.security.krb5.KrbApReq.<init>(KrbApReq.java:149)
    at java.security.jgss/sun.security.jgss.krb5.InitSecContextToken.<init>(InitSecContextToken.java:139)
    at java.security.jgss/sun.security.jgss.krb5.Krb5Context.acceptSecContext(Krb5Context.java:832)
    ... 269 more
```

Sur les ADs, la valeur de chiffrement par défaut est RC4-HMAC. Or La mise à jour passe cette valeur par défaut sur AES (128, 256 ,etc.).

CAS faisant du rc4-hmac, il nous faut passer en AES256-CTS.

L'équipe Windows nous a généré un nouveau keytab chiffré avec le bon algorithme et nous avons modifié le fichier krb5.conf :

default_tkt_enctypes = aes256-cts
default_tgs_enctypes = aes256-cts

Pensez à bien passer la commande **klist purge** sur les postes clients Windows (**kdestroy** pour linux)... sinon, il faut attendre la date de renouvellement des tickets KERBEROS (7 jours par défaut).

Pas besoin de redémarrer CAS 🙂