CARTES CPS

PKCS#11 RECHERCHE CLE

Identification du document	
Référence ANS	ANS_CPSV4_0015_Note_PKCS11_Find_Key.docx
Date de création	29/01/2025
Date de dernière mise à jour	29/01/2025
Etat	En cours / A vérifier / A valider / **Validé**
Rédaction (R)	ANS
Version	**V 1.0**
Vérification	
Validation finale (A)	
Classification	**Diffusion restreinte**

Historique du document			
Version	**Date**	**Auteur**	**Commentaires**
V1.0	18/12/2024	ANS/GIE	Création du document

Sommaire

1 Introduction

L'objet de cette note est de décrire différentes méthodes permettant de récupérer, au sein de la carte CPS, l'objet clé privé de signature ou d'authentification afin de réaliser des opérations cryptographiques.

Toute carte CPS a au moins deux paires de bi-clé :

USAGE	LABEL CERTIFICAT	LABEL CLE PRIVEE
AUTHENTIFICATION	Certificat d'Authentification CPS	CPS_PRIV_AUT
SIGNATURE	Certificat de Signature CPS	CPS_PRIV_SIG

Les labels présentés ci-dessus sont donc fixes et ne changent pas d'une version de carte CPS à l'autre.

On présente dans cette note 2 méthodes pour récupérer l'objet clé privée que l'on souhaite utiliser.

2 Recherche avec ID

La méthode est décrite dans le document

[1] ANS_PTS_PSCE_MP_Cryptolib_CPS_v5_Manuel_de_programmation_20131016_v1.5.0.pdf

Elle consiste dans un premier temps à récupérer l'ID de la clé. Pour cela on effectue une recherche de certificat avec le label souhaité (Signature ou Authentification, Voir Chapitre 1). Cette recherche est décrite dans le chapitre 6.5.2 de [1]

```
CK_RV rc = CKR_OK;
/* classe d'objet à rechercher */
CK_OBJECT_CLASS objClass = CKO_CERTIFICATE;
CK_OBJECT_HANDLE hCert;
/* l'objet à rechercher est public CKA_PRIVATE->CK_FALSE */
CK_BBOOL bFalse = CK_FALSE;
/* l'objet à rechercher est token CKA_TOKEN->CK_TRUE */
CK_BBOOL bTrue = CK_TRUE;
/* Nombre maximum d'objets à récupérer */
CK_ULONG ulMaxObjectCount = 1;
/* Valeur binaire de l'objet */
CK_BYTE *value = NULL;
/* Longueur de la valeur binaire de l'objet */
CK_ULONG lenValue = sizeof(value);
/* Label de l'objet */
char label[] = "Certificat d'Authentification CPS";  ➔ ici recherche
certificat d'authentification
/* Template de recherche */
CK_ATTRIBUTE searchTemplate[]=
{
{CKA_CLASS, &objClass, sizeof(objClass)},
{CKA_TOKEN, &bTrue, sizeof(bTrue)},
{CKA_PRIVATE, &bFalse, sizeof(bFalse)},
{CKA_LABEL, label, strlen(label)}
};
/* Template de récupération de donnée */
CK_ATTRIBUTE templateAttr[] =
{
{CKA_VALUE, NULL_PTR, 0},
{CKA_ID, NULL, 0}
};
CK_ULONG searchTemplateSize =
sizeof(searchTemplate)/sizeof(CK_ATTRIBUTE);
CK_ULONG templateAttrSize = sizeof(templateAttr)/sizeof(CK_ATTRIBUTE);
rc = (*pFunctionList->C_FindObjectsInit)(hSession, searchTemplate,
searchTemplateSize);
….
```

Une fois l'ID récupéré, on réalise une seconde recherche comme indiqué au chapitre 6.6.1 de [1]

```
CK_RV rc = CKR_OK;
/* classe d'objet à rechercher */
CK_OBJECT_CLASS objClass = CKO_PRIVATE_KEY;
CK_KEY_TYPE keyType = CKK_RSA;
CK_OBJECT_HANDLE hKey;
CK_ULONG ulMaxObjectCount = 1;
CK_BYTE_PTR pSignature = NULL;
CK_ULONG ulSignatureLen;
/* Template de recherche */
CK_ATTRIBUTE searchTemplate[]=
{
{CKA_CLASS, &objClass, sizeof(objClass)},
{CKA_KEY_TYPE, &keyType, sizeof(keyType)},
{CKA_ID, &cka_id_keyAuth, sizeof(cka_id_keyAuth)}, ➔ cet id vient de la recherche précédente (certificat)
};
CK_ULONG searchTemplateSize =
sizeof(searchTemplate)/sizeof(CK_ATTRIBUTE);
rc = (*pFunctionList->C_FindObjectsInit)(hSession, searchTemplate,
searchTemplateSize);
…
```

3 Recherche avec LABEL

Cette seconde méthode utilise le label de la clé privée et permet de récupérer immédiatement la clé souhaitée.

```
CK_RV rc = CKR_OK;
/* classe d'objet à rechercher */
CK_OBJECT_CLASS objClass = CKO_PRIVATE_KEY;
char label[] = "CPS_PRIV_AUT";  ➔ ici recherche clé privée
d'authentification
CK_OBJECT_HANDLE hKey;
CK_ULONG ulMaxObjectCount = 1;
CK_BYTE_PTR pSignature = NULL;
CK_ULONG ulSignatureLen;
/* Template de recherche */
CK_ATTRIBUTE searchTemplate[]=
{
{CKA_CLASS, &objClass, sizeof(objClass)},
{CKA_LABEL, label, strlen(label)}
};
CK_ULONG searchTemplateSize =
sizeof(searchTemplate)/sizeof(CK_ATTRIBUTE);
rc = (*pFunctionList->C_FindObjectsInit)(hSession, searchTemplate,
searchTemplateSize);
…
```

Dans cette méthode, même l'attribut KEY TYPE (RSA) n'est pas fourni car il permet de se prémunir d'un changement de type de clé de la carte (passage à ECC par exemple …). Cette recherche repose sur le fait que les labels de la CPS sont fixes.

4 LOG

4.1.1 Introduction

Ci-dessous des exemples de résultats (LOG) effectués sur des cartes CPS V3 et CPS V4 (méthode d'un seul FIND OBJECT avec label). Exemples réalisés en PYTHON.

Extrait du script PYTHON effectuant la recherche d'objets :

```
def getPrivateCPS(self,label):
        objects =
self.session.findObjects([(CKA_CLASS,CKO_PRIVATE_KEY),(CKA_PRIVATE,True),
(CKA_LABEL,label)])
        if len(objects) == 0:
            print("--> Error key not found")
        elif len(objects) != 1:
            print("--> More than 1 key found")
        else:
            result = objects[0].to_dict()
            # Print RSA key found .... print LABEL and ID
            print("RSA label = ",result["CKA_LABEL"])
            print("RSA_ID    = ",result["CKA_ID"])
            return objects[0]
        return None
    ......
    print("\n")
    print("Get RSA AUTH key:\n================\n")
    self.getPrivateCPS("CPS_PRIV_AUT")
    print("\n")
    print("Get RSA SIGN key:\n================\n")
    self.getPrivateCPS("CPS_PRIV_SIG")
    ......
```

4.1.2 Recherche avec LABEL / CPS 3

```
>>> %Run cps-script02.py
Infos slot 0:
=============
firmwareVersion: 0.00
flags: CKF_TOKEN_PRESENT, CKF_REMOVABLE_DEVICE, CKF_HW_SLOT
hardwareVersion: 0.00
manufacturerID:
slotDescription: Alcorlink USB Smart Card Reader 0

Infos token slot 0:
===================
firmwareVersion: 0.0
flags: CKF_RNG, CKF_LOGIN_REQUIRED, CKF_USER_PIN_INITIALIZED,
CKF_TOKEN_INITIALIZED
hardwareVersion: 0.0
label: CPS3v3-2900638158
manufacturerID: ASIP SANTE
model: IAS ECCNone

Get RSA AUTH key:
================
RSA label =  CPS_PRIV_AUT
RSA_ID    =  (232, 40, 189, 8, 15, 128, 37, 0, 0, 1, 255, 0, 16, 2)

Get RSA SIGN key:
================
RSA label =  CPS_PRIV_SIG
RSA_ID    =  (232, 40, 189, 8, 15, 128, 37, 0, 0, 1, 255, 0, 16, 1)
[Event-ENDS]No Error.

>>>
```

4.1.3 Recherche avec LABEL / CPS 4

```
>>> %Run cps-script02.py

Infos slot 0:
=============
firmwareVersion: 0.00
flags: CKF_TOKEN_PRESENT, CKF_REMOVABLE_DEVICE, CKF_HW_SLOT
hardwareVersion: 0.00
manufacturerID:
slotDescription: Alcorlink USB Smart Card Reader 0

Infos token slot 0:
===================
firmwareVersion: 0.0
flags:     CKF_RNG,     CKF_LOGIN_REQUIRED,     CKF_USER_PIN_INITIALIZED,
CKF_TOKEN_INITIALIZED
hardwareVersion: 0.0
label: CPS4v1-3100603745
manufacturerID: ANS
model: IAS ECCNone

Get RSA AUTH key:
================
RSA label =  CPS_PRIV_AUT
RSA_ID    =  (232, 40, 189, 8, 15, 128, 37, 0, 0, 1, 255, 0, 16, 32)

Get RSA SIGN key:
================
RSA label =  CPS_PRIV_SIG
RSA_ID    =  (232, 40, 189, 8, 15, 128, 37, 0, 0, 1, 255, 0, 16, 16)
 [Event-ENDS]No Error.

>>>
```