Sistema di controllo della Tracciabilità dei Rifiuti SISTRI

Interoperabilità SISTRI
Specifiche tecniche per l'utilizzo della firma elettronica con il Soft Token PKCS#11

Prot. N.: SISTRI-TN_SIS-001 FE
Versione: 1.0
Data: 28/09/2010

MINISTERO DELL'AMBIENTE
E DELLA TUTELA DEL TERRITORIO E DEL MARE

STORIA DEL DOCUMENTO

VER.	DATA	DESCRIZIONE
1.0	28/09/2010	Prima Edizione

Sommario

1 Architettura

1.1 Descrizione generale

Soft Token PKCS#11 è un modulo software parte del token SISTRI.

Nel contesto del token SISTRI, il modulo Soft Token PKCS#11 si occupa di veicolare le credenziali di firma elettronica e autenticazione forte basate su PKI, offrendo uno "store" e una libreria per tali meccanismi, indipendente da specifici applicativi e sistemi operativi.

Soft Token PKCS#11 opera simulando le funzionalità di un token crittografico PKCS#11, nella fattispecie è totalmente equivalente ad una SmartCard di firma elettronica.

1.2 Diagramma dell'architettura

La figura seguente mostra nell'insieme i moduli del software crittografico, gli interfacciamenti fra i moduli sono realizzati tramite interfacce standard (PKCS#11) garantendo, di conseguenza, una possibile sostituzione dei moduli fronte di possibili evoluzioni tecnologiche.

1.3 Caratteristiche Funzionali

Soft Token PKCS#11 è in grado di operare come un dispositivo di firma elettronica standard PKCS#11 riproducendo tutte le funzionalità di un token di firma elettronica standard.

Le funzionalità sono implementate utilizzando le interfacce e le strutture dati descritte nella documentazione ufficiale di riferimento (http://www.rsa.com/rsalabs/node.asp?id=2133) nei profili funzionali "***RSA Asymmetric Client Signing Profile***" e "***RSA Asymmetric Acceleration Profile***": oltre ai meccanismi di crittografia asimmetrica e firma elettronica, sono anche implementate le funzioni necessarie a supportate le seguenti funzionalità di alto livello:

- Generazione di coppie di chiavi RSA (ad esempio finalizzata alla realizzazione da parte del client di procedure di enrollment secondo gli standard);
- Funzione di firma e di decifratura con chiave privata protetta da PIN (utile a proteggere file e documenti riservati);
- Funzione di firma e di decifratura con chiave privata protetta da PIN (finalizzata all'autenticazione, ad esempio di tipo SSL, VPN, etc.);
- Funzioni di cifratura e decifratura con algoritmi DES, 3DES e AES.

Il software non richiede alcuna installazione nei sistemi sui quali opera né richiede all'utente di disporre dei privilegi di amministratore del sistema.

1.4 Architetture Hardware/Software Supportate

Nel token SISTRI, Soft Token PKCS#11 è disponibile in 3 distinte versioni, per altrettante piattaforme hardware/software:

- Microsoft Windows, versioni "XP", "Vista", "Seven (7)";
- Linux, compatibile con tutte le distribuzioni basate su kernel 2.6.x e processori compatibili IA32;
- Apple Mac OS X, nelle versioni 10.4.x, 10.5.x, 10.6.x su processori Intel.

Soft Token PKCS#11 è compilato a 32 bit: questo lo rende compatibile con applicazioni a 32 bit eseguite su sistemi operativi a 32 e 64 bit.

Il token può essere impiegato indifferentemente su tutti i sistemi operativi supportati, è anche possibile alternarne l'uso su computer differenti e sistemi operativi differenti senza che questo alteri in alcun modo le sue caratteristiche.

Le librerie da interfacciare possono essere ricercate all'interno dei seguenti percorsi:

Windows:

[unità-logica]:\sistri\DigitalID\SoftTokenEngine.dll

Linux :

[mount-point-chiavetta]/sistri/DigitalID/libSoftTokenEngine.so

MacOS X:

[mount-point-chiavetta]/sistri/DigitalID/libSoftTokenEngine.dylib

2 Caratteristiche specifiche del software

Il P11_Datastore implementa la crittografia ed esporta una interfaccia secondo lo standard PKCS#11.

Il modello presentato in questo paragrafo rappresenta il contesto applicativo di riferimento per la soluzione software per la parte client del TOKEN USB. Il modello è applicabile nel caso di utilizzo di una soluzione completamente software:

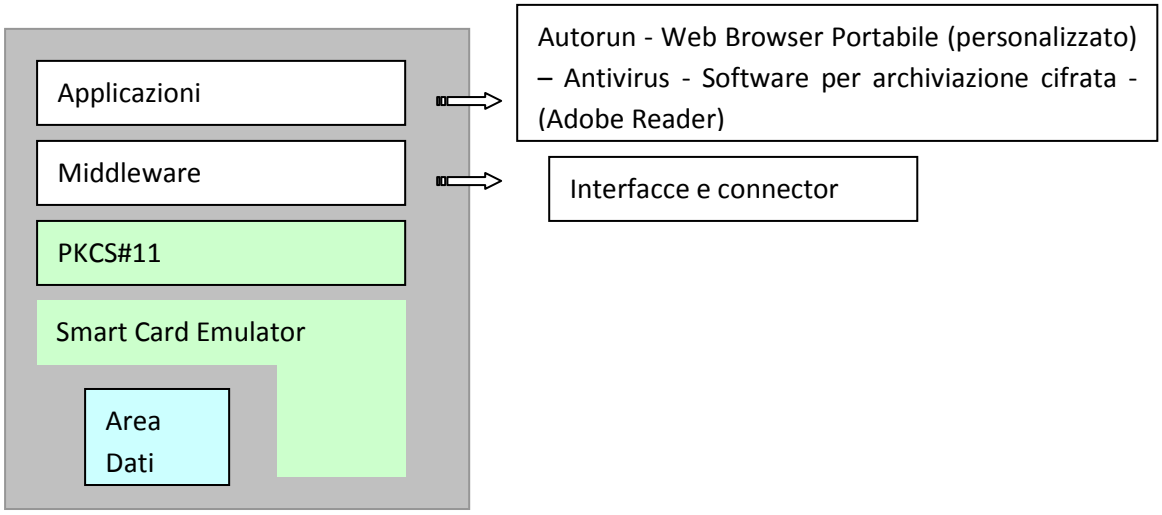

figura 1 :modello architettura software soft token

La figura esemplifica le due peculiarità maggiori del TOKEN Software:

1. Il token emula quindi una smart card per l'autenticazione al SISTRI e firma in rete. La gestione delle funzioni crittografiche è realizzata tramite l'interfaccia standard PKCS#11.
2. Il Token non richiede alcun driver kernel-mode o driver di lettore di smart card, questa caratteristica lo rende utilizzabile senza richiedere una installazione permanente nel sistema e accessibile anche dalle applicazioni impiegate da utenti privi di privilegi di amministrazione del sistema.

2.1 Profilo di conformità standard PKCS#11

2.1.1 Compatibilità

Il prodotto espone le funzionalità più utilizzate dagli applicativi di firma elettronica e cifratura, segue un elenco più dettagliato diviso per tipologia di funzione.

Per una descrizione più completa della funzione supportata viene descritta nel documento ufficiale RSA PKCS#11 v2.20: Cryptographic Token Interface Standard, ftp://ftp.rsasecurity.com/pub/pkcs/pkcs-11/v2-20/pkcs-11v2-20.pdf il numero di pagina riferita a questo documento viene segnalato per ogni singola funzione.

2.1.2 General-purpose functions

C_Initialize (pag. 102), la funzione è 100% compatibile. L'eventuale fallimento funzione è unicamente legato al tentativo di copiatura dei file di sistema del token.

C_Finalize (pag. 104).

C_GetFunctionList (pag. 106).

C_GetInfo (pag. 105).

2.1.3 Slot and Token Management Functions

C_GetSlotList (pag. 106).

C_GetSlotInfo (pag. 108).

C_GetTokenInfo (pag. 109).

C_GetMechanismList (pag. 111).

C_GetMechanismInfo (pag. 109).

C_InitToken (pag. 113)

C_InitPIN (pag. 115).

C_SetPIN (pag. 116).

2.1.4 Session Management Functions

C_OpenSession (pag. 117).

C_CloseSession (pag. 118).

C_CloseAllSessions (pag. 120).

C_GetSessionInfo (pag. 120)

C_Login (pag. 125)

C_Logout (pag. 127)

2.1.5 Object Management Functions

C_CreateObject (pag. 128), è possibile creare oggetti Certificato, Chiavi pubbliche e provate e Secret keys.

C_DestroyObject (pag. 131).

C_GetObjectSize (pag. 132).

C_GetAttributeValue (pag. 133), è possibile ottenere i seguenti attributi:

CKA_CLASS, CKA_LABEL, CKA_TOKEN, CKA_PRIVATE, CKA_ID, CKA_SENSITIVE, CKA_ENCRYPT, CKA_DECRYPT, CKA_WRAP, CKA_UNWRAP, CKA_SIGN, CKA_VERIFY, CKA_VALUE, CKA_CERTIFICATE_TYPE, CKA_ISSUER, CKA_SERIAL_NUMBER, CKA_SUBJECT, CLA_START_DATE, CKA_END_DATE, CKA_KEY_TYPE, CKA_MODULUS, CKA_MODULUS_BITS, CKA_PUBLIC_EXPONENT, CKA_PRIVATE_EXPONENT, CKA_NEVER_EXTRACABLE, CKA_MODIFIABLE, CKA_EXTRACTABLE, CKA_LOCAL.

C_SetAttributeValue (pag. 135) è possibile scrivere i seguenti attributi:

CKA_LABEL, CKA_ID, CKA_SUBJECT.

C_FindObjectsInit (pag. 136), è possibile cercare oggetti per i seguenti attributi:

CKA_CLASS, CKA_CERTIFICATE_TYPE, CKA_KEY_TYPE, CKA_LABEL, CKA_SUBJECT, CKA_ID, CKA_VALUE.

C_FindObjects (pag. 137).

C_FindObjectsFinal (pag. 138).

2.1.6 Encryption Functions

C_EncryptInit (pag. 139) supportati CKM_AES_ECB, CKM_AES_CBC, CKM_DES3_CBC_PAD.

C_Encrypt (pag. 140) supportati CKM_AES_ECB, CKM_AES_CBC, CKM_DES3_CBC_PAD.

2.1.7 Decryption Funcions

C_DecryptInit (pag. 144) supportati CKM_AES_ECB, CKM_AES_CBC, CKM_DES3_CBC_PAD.

C_Decrypt (pag. 145) supportati CKM_AES_ECB, CKM_AES_CBC, CKM_DES3_CBC_PAD.

2.1.8 Message Digest Functions

C_DigestInit (pag. 148) sono implementati i seguenti algoritmi: CK_MD2, CKM_MD5, CKM_SHA1.

C_Digest (pag. 148) sono implementati i seguenti algoritmi: CK_MD2, CKM_MD5, CKM_SHA1.

C_DigestUpdate (pag. 150) sono implementati i seguenti algoritmi: CK_MD2, CKM_MD5, CKM_SHA1.

C_DigestFinal (pag. 151) sono implementati i seguenti algoritmi: CK_MD2, CKM_MD5, CKM_SHA1.

2.1.9 Signing Functions

C_SignInit (pag. 157).

C_Sign (pag. 158).

2.1.10 Functions for Verifying Signatures

C_VerifyInit (pag. 157).

C_Veirfy (pag. 158).

2.1.11 Key Management Functions

C_GenerateKey (pag. 175), sono supportati: CKM_AES_KEY_GEN, CKM_DES3_KEY_GEN.

C_GenerateKeyPair (pag. 176) supportato: CKM_RSA_PKCS_KEY_PAIR_GEN.

2.1.12 Random Number Generation Functions.

C_SeedRandom (pag. 184).

C_GenerateRandom (pag. 184).

2.2 Comportamenti specifici extra standard

Il SoftToken implementa funzione di crittografia, e conservazione sicura di chiavi e dati multiutente.

Al fine di poter utilizzare un token da parte di molteplici persone fisiche il token implementa particolari funzioni crittografiche che provvedono a generare dei “p11_datastore”.

Non esistono funzioni a livello di libreria per selezionare l’utente per il quale svolgere le funzioni di firma cifra etc. Nel caso il token sia stato personalizzato in modalità multiutente la libreria provvederà a richiedere interattivamente all’utente stesso la propria USERID dopo la richiesta di passphrase.