Uso pratico di Inline::Java, j4sign & Bouncy Castle Crypto APIs

Esempio di applicazione Apache/Mason
per la verifica delle firme digitali
e delle marcature temporali

L'applicazione è utilizzata nell'ambito del
Progetto “Gestione Analisi Chimiche”
In uso presso la Provincia Autonoma di Trento

Perl

HTML::Mason

Apache

Inline::Java

j4sign

Bouncy Castle

In Italia dal 2011 le applicazioni che implementano meccanismi di firma digitale a valore legale devono supportare SHA 256 e il recente formato per imbustare documenti con marcatura temporale (RFC 5544).

In ambiente Perl non è tuttora disponibile del codice che permetta di firmare e marcare temporalmente secondo le norme in vigore, né tanto meno verificarle.

A inizio luglio 2011, in corrispondenza dell'entrata in vigore delle nuove norme, si è reso necessario trovare una soluzione per aggiornare il codice di verifica firma utilizzato nel progetto di gestione Analisi Chimiche.

?

In quella data (e probabilmente anche oggi) non esistevano progetti Open Source di verifica di firma e marcatura temporale se non integrando in **j4sign** una nuova API che implementa la RFC 5544 (imbustamento del documento firmato e della marcatura temporale).

j4sign è un progetto finanziato dal Comune di Trento per la firma digitale che utilizza le classi di **"Legion of the Bouncy Castle"**.

Per nostra fortuna nella versione 1.46 era disponibile una nuova classe per manipolare le buste nel formato rfc5544.

j4sign - http://j4sign.sourceforge.net
Bouncy Castle - http://www.bouncycastle.org

Ma come integrare le classi Java in ambiente Perl?

Inline::Java

Con Inline::Java è stato possibile integrare una Java Virtual Machine in Apache chiamando direttamente in Perl le classi Java.

Inline::Java - http://search.cpan.org/dist/Inline-Java/Java.pod

Perl, Mason & Inline::Java

Relatore
Guido Brugnara


```
use Inline (
   Java => 'STUDY',
   SHARED_JVM => 1,
   START_JVM => 0,
   PORT => 7893,
   AUTOSTUDY => 1,
   # DEBUG => 2,
   DIRECTORY => $r->dir_config('VerifyService_directory'),
   STUDY => ['it.trento.comune.j4sign.verification.servlet.VerifyService'],
   CLASSPATH => $r->dir_config('VerifyService_classPath')
 );

my $VerifyService = new HTML::Mason::Commands::it::trento::comune::\
                                                  j4sign::verification::servlet::VerifyService(
   $r->dir_config('VerifyService_confDir'),
   $r->dir_config('VerifyService_cnipaDir'),
   $r->dir_config('VerifyService_cnipaCa'),
   $r->dir_config('VerifyService_cnipaRoots'),
   $r->dir_config('VerifyService_fingerprintDigitPA')
 );
```

Attivazione della JVM

Studio della classe

Istanzia la classe

Parsing del file nel formato TSD (RFC 5544)

```
my $res = $VerifyService->parseTSD($sfile);
$fileP7M = "$sfile.p7m";

my $is_ok = $VerifyService->validateTSD($fileP7M)

my $tokens = $VerifyService->tokensTSD;
foreach my $token (@$tokens){
    my $certificates = $token->getCertificates->getMatches(undef)->toArray;
    foreach my $cert (@$certificates){
        $m->out('Timestamp emesso da: '.$cert->getIssuer->toString);
    }
    $timestampDate = $token->getTimeStampInfo->getGenTime;
    last;
}
```

Validazione

Array dei token
(interessa il primo)

Data della marcatura temporale

Il codice completo e le istruzioni per la compilazione
sono disponibili all'indirizzo web:

https://www.leader.it/Blog/Uso_pratico_di_InlineJava_j4sign__Bouncy_Castle_Crypto_APIs

Il codice è distribuito con licenza Affero G.P.L. v1

http://www.affero.org/oagpl.html

Uso pratico di Inline::Java, j4sign & Bouncy Castle Crypto APIs

Grazie!

:-)

IFASE

Leader.IT NETWORK
www.leader.it
info@leader.it
Rif. ing Guido Brugnara

Committente:
PROVINCIA AUTONOMA DI TRENTO
Agenzia per la depurazione
Via Pozzo, 6
38122 TRENTO
http://www.adep.provincia.tn.it/

Aziende coinvolte:
IFASE - Informatic, Facility Automation and Software for Environment
Strada della Pozzata, 41
38123 TRENTO
http://www.ifase.it/
info@ifase.it