Implementazione di un Protocollo di Firma Elettronica Avanzata basato su SPID

Francesco Buccafurri, Gianluca Lax e Antonia Russo
Università degli Studi "Mediterranea" di Reggio Calabria
bucca@unirc.it, lax@unirc.it, antonia.russo@unirc.it

Abstract. Il quadro normativo europeo che regola identità digitale e firme elettroniche prevede, tra le altre forme di firma elettronica, la firma elettronica avanzata. Sebbene tale istituto sia presente anche nell'ordinamento giuridico italiano da diversi anni, si avverte certamente la necessità di definire soluzioni tecniche convincenti che implementino una firma elettronica avanzata in modo sicuro, portabile, e facilmente verificabile. Recentemente è stato proposto un approccio che sfrutta il sistema di identità digitale pubblica per la realizzazione di un protocollo di firma elettronica avanzata. La proposta appare convincente anche perché fa confluire diversi strumenti in uno rappresenta una significativa semplificazione per cittadini e imprese. In questo articolo offriamo alcuni spunti di approfondimento di carattere implementativo evidenziando la realizzabilità e l'efficacia della proposta.

Keywords: informatica nella pubblica amministrazione, identità digitale, eIDAS, CAD.

1 Introduzione

Il recente nel Regolamento Europeo eIDAS (electronic IDentification Authentication and Signature) n° 910/2014 mira a fornire una base normativa a livello comunitario per i servizi fiduciari e i mezzi di identificazione elettronica degli stati membri. Tra le altre cose, questo regolamento istituisce un quadro giuridico per le firme elettroniche, definendo le norme e procedure per le firme elettroniche in cui sono stabilite le condizioni per l'interoperabilità a livello comunitario. La firma elettronica è definita come un insieme dei dati in forma elettronica, allegati oppure connessi tramite associazione logica ad altri dati elettronici, utilizzati come metodo di identificazione informatica. Questo è il tipo più debole di firma [19], perché non include meccanismi di autenticazione o di integrità. Un tipo di firma più sicura è la firma elettronica qualificata (FEQ), definita come una firma elettronica ottenuta attraverso una procedura informatica che garantisce la connessione univoca al firmatario, creata con mezzi sui quali il firmatario può conservare un controllo esclusivo e collegata ai dati ai quali si riferisce in modo da consentire di rilevare se i dati stessi siano stati successivamente modificati, che sia basata su un certificato qualificato e realizzata mediante un dispositivo sicuro per la creazione della firma.

Il legislatore europeo e quello nazionale hanno individuato l'opportunità di prevedere una forma di firma elettronica che abbia minori vincoli (anche procedurali) della firma elettronica qualificata, ma che offra allo stesso tempo idonee garanzie di sicurezza in termini di legame tra firma e contenuto, capace di rilevare ogni sua modifica, e tra firmatario e mezzi con i quali, in maniera esclusiva, il firmatario genera la firma. Stiamo parlando della firma elettronica avanzata (FEA), che rispetto alla (FEQ) non richiede che sia basata su un certificato qualificato e realizzata mediante un dispositivo sicuro per la creazione della firma, quale l'apparato strumentale usato per la creazione della firma elettronica.
A tutt'oggi tuttavia non vi sono esperienze significative di applicazione di questo istituto al settore della PA, settore per il quale principalmente la FEA è stato pensato. Di fatto l'unica implementazione di FEA è quello della firma grafometrica che però presenta diversi svantaggi tra i quali non trascurabili quelli legati al fatto che si tratta di soluzioni chiuse e controllate dai vendor. Inoltre, il legame tra contenuto del documento e firma è realizzato attraverso metodi crittografici ma memorizzato localmente nel punto in cui la firma è generata e non vi è modo di rendere il documento esportabile in modo tale che la sua FEA possa essere verificata in qualsiasi altro punto. Anche la portabilità in fase di generazione è fortemente compromessa.
Alla luce di queste osservazioni, è evidente quanto possa essere rilevante proporre protocolli di FEA che siano più efficaci e che abbiano caratteristiche di portabilità e verificabilità universale [11,12,14].
Nel 2016, nel lavoro [13], è stato definito un protocollo di FEA che utilizza il sistema di identità digitale pubblica SPID [10]. In linea di principio, l'approccio è applicabile ad un qualsiasi sistema di identità digitale pubblica compatibile con il quadro regolatorio e tecnologico previsto in eIDAS e nella normativa correlata [5,15,16]. In questo articolo viene ripresa ed approfondita la proposta presentata in [13] al fine di rendere chiari ulteriori dettagli implementativi e dimostrare la realizzabilità pratica della soluzione. È interessante osservare che l'iniziale idea di realizzare un protocollo di firma elettronica avanzata attraverso SPID, presentata nel 2016 in [13] è stata di fatto adottata nel 2018 nella definizione di una soluzione pratica dall'istituto INPS, come descritto in [9]. Ciò fa pertanto ritenere che la direzione immaginata nella proposta iniziale sia effettivamente percorribile ed ulteriormente esportabile verso domini generali. Scopo di questo lavoro è pertanto offrire nuovi spunti che rafforzano la proposta e il suo percorso di applicazione.

La struttura del lavoro è la seguente. Nella Sezione 2 viene analizzato il meccanismo di autenticazione in SPID, con riferimento alle regole tecniche e alle definizioni del sistema stesso. Nella Sezione 3 è presentata la nostra proposta di firma elettronica avanzata basata su SPID. Infine, nella Sezione 4, vengono tratte le conclusioni.

2 SPID

In questa sezione descriviamo dettagliatamente come avviene l'autenticazione tramite SPID, in quanto questo protocollo verrà modificato in modo da implementare un sistema di firma elettronica avanzata.
Il sistema pubblico di identità digitale SPID è stato progettato in conformità al Regolamento eIDAS (electronic IDentification Authentication and Signature) e consente ai cittadini italiani e alle imprese di accedere ai servizi online della pubblica amministrazione e dei privati aderenti con un'identità digitale unica. Descriviamo nel seguito il meccanismo di autenticazione basato su SPID attuato quando un utente deve accedere a un servizio fornito da un *Service Provider*: lo scambio dei messaggi previsti da questo meccanismo è mostrato in Figura 1. L'utente utilizza un browser (*User Agent*) e invia

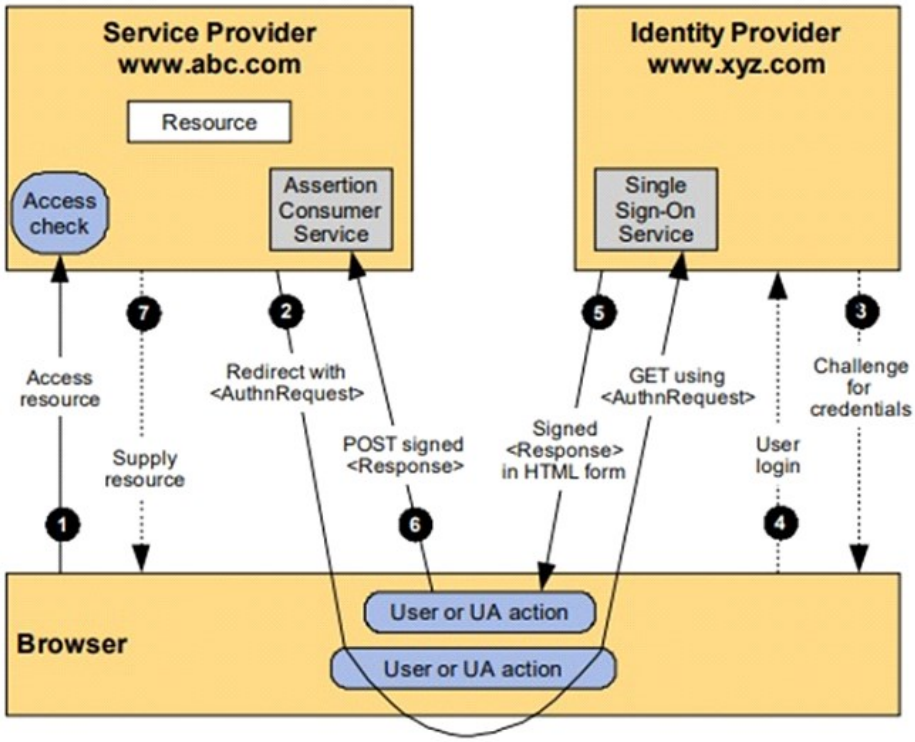

Fig. 1. Autenticazione in SPID.

al *Service Provider* una richiesta di accesso al servizio (**Step 1**). Quindi, il *Service Provider* risponde all'*User Agent* con una richiesta di autenticazione da inoltrare all'*Identity Provider* (**Step 2**). La richiesta di autenticazione segue lo standard SAML [1] ed è basata sul costrutto XML <AuthnRequest>: può essere inoltrata da un *Service Provider* all'*Identity Provider* usando il *binding HTTP Redirect* o il *binding* http POST. La relativa risposta SAML è basata sul costrutto <Response> e può invece essere inviata dall'*Identity Provider* al *Service Provider* solo tramite il binding HTTP POST. L'*AuthnRequest* deve essere conforme allo standard SAML v2.0 [3,8] ed è così definito [4]:

Definizione 1. La richiesta di autenticazione AuthnRequest contiene i seguenti campi:

- l'attributo **ID**, chè è un *Universally Unique Identifier* (UUID), tipicamente una combinazione origine+timestamp (quest'ultimo generato con una precisione di almeno un millesimo di secondo per garantire l'univocità);
- l'attributo **Version**, che rappresenta la versione della specifica SAML adottata (deve essere almeno la versione "2.0");
- l'attributo **IssueInstant,** che indica l'istante di emissione della richiesta, in formato UTC (per esempio: "2018-09-08T18:04:15.531Z");
- l'attributo **Destination**, ossia l'indirizzo (URI) dell'*Identity Provider* a cui è inviata la richiesta, come risultante nell'attributo entityID presente nel metadata IdP dell'*Identity Provide*r a cui viene inviata la richiesta;
- l'attributo **ForceAuthn,** nel caso in cui si richieda un livello di autenticazione superiore a SPIDL1 (cioè, SPIDL2 o SPIDL3);
- l'attributo **AssertionConsumerServiceIndex**, che riporta un indice posizionale facente riferimento ad uno degli elementi <AttributeConsumingService> presente nei metadata del *Service Provider*: Tale elemento indica, mediante l'attributo Location, l'URL a cui inviare il messaggio di risposta alla richiesta di autenticazione e mediante l'attributo Binding, il binding da utilizzare (quest'ultimo valorizzato è obbligatoriamente un "urn:oasis:names:tc:SAML:2.0:bindings:HTTP-POST");
- in alternativa al precedente attributo possono essere presenti:
    - l'attributo **AssertionConsumerServiceURL** che indica l'URL a cui inviare il messaggio di risposta alla richiesta di autenticazione (l'indirizzo deve coincidere con quello del servizio riportato dall'elemento <**AssertionConsumingService**> presente nei metadata del *Service Provider*);
    - l'attributo **ProtocolBinding**, identificante il binding da utilizzare per inoltrare il messaggio di risposta, valorizzato con "urn:oasis:names:tc:SAML:2.0:bindings: HTTP-POST";
- un attributo facoltativo **<AttributeConsumingServiceIndex>** riportante un indice posizionale in riferimento alla struttura presente nei metadata del *Service Provider*, atta a specificare gli attributi che devono essere presenti nell'*assertion* prodotta. Nel caso l'attributo fosse assente, l'*assertion* prodotta non riporterà alcuna attestazione di attributo;

• può essere presente l'elemento **<Subject>** a indicare il soggetto per cui si chiede l'autenticazione, in cui deve comparire l'elemento <**NameId**> atto a qualificare il soggetto, in cui sono presenti i seguenti attributi:

  • **Format** che deve assumere un valore di tipo "`urn:oasis:names:tc:SAML:1.1: nameidformat:unspecified`";

  • **NameQualifier** che qualifica il dominio a cui afferisce tale valore (URI).

• deve essere presente l'elemento **<Issuer>**, valorizzato con l'attributo entityID riportato nel corrispondente SP metadata, che individua il Service Provider emittente;
• un elemento **<NameIDPolicy>**, che definisce i formati identificativi del nome supportati dall'*Identity Provider*;
• un elemento opzionale **<Conditions>**, che specifica il tempo di validità;
• un elemento **<RequestedAuthnContext>**, che indica la robustezza delle credenziali richieste;
• un elemento facoltativo **<Signature>**, contenente la firma sulla richiesta apposta dal Service Provider.

Se la richiesta ricevuta è valida, l'*Identity Provider* esegue l'autenticazione dell'utente (**Step 3** e **4**). In caso di autenticazione utente avvenuta con successo, l'*Identity Provider* prepara l'*assertion*, definita come segue.

Definizione 2. L'*Assertion* contiene i seguenti campi:
• Un **ID,** generalmente ottenuto da una combinazione di origine e data/ora, ad esempio `Assertion- uuidae7136e4-0118-18d8-999dc ff934 ae63db`;
• un attributo **Version**, che indica la versione di SAML del messaggio;
• un attributo **IssueInstant**, che specifica l'istante in cui è stata emessa la richiesta;
• un elemento **<Subject>**, che identifica l'utente autenticato e deve contenere un elemento <**SubjectConfirmation**>, con un attributo **InResponseTo**, che si riferisce all'ID della corrispondente `AuthnRequest`;
• un elemento **<Issuer>**, che specifica l'`EntityID` dell'`Identity Provider`;
• un elemento **<Conditions>**, che definisce l'intervallo temporale di validità;
• un elemento **<AuthStatement>**, che è la descrizione del contesto dell'autenticazione;
• un elemento **<AttributeStatement>**, che contiene il codice di identificazione SPID dell'utente autenticato;
• un elemento **<Signature>**, ossia la firma dell'`Identity Provider` dell'asserzione.

Conclusa la fase di autenticazione, l'*Identity Provider* costruisce un messaggio di tipo `<Response>`, che viene inserito in un form HTML come campo nascosto di nome "`SAMLResponse`".
L'*Identity Provider* invia tale form HTML al browser dell'utente (**Step 5**).
Il browser dell'utente elabora quindi la risposta HTTP e inoltra, attraverso HTTP POST, la `<Response>` al Service Provider (**Step 6**).

Definizione 3. Il messaggio di risposta (`Response`) è definito come:
- un ID attributo univoco;
- un attributo **`Version`**, che indica la versione di SAML del messaggio;
- un attributo **`IssueInstant`**, che specifica l'istante in cui è stata emessa la richiesta;
- un attributo **`InResponceTo`**, contenente il valore dell'ID dell'attributo di `AuthnRequest`;
- un attributo **`Destination`**, l'URI a cui deve essere inviata la risposta;
- un elemento **<`Status`>**, che specifica il risultato della richiesta (ad es. successo);
- un elemento **<`Issuer`>**, che riporta l'`EntityID` di Identity Provider;
- un elemento **<`Assertion`>** (già descritto nella Definizione 2);
- un elemento opzionale **<`Signature`>**, ossia la firma del Identity Provider (solo nel caso di binding POST HTTP).

Le caratteristiche dell'*Identity Provider* devono essere definite attraverso metadata conformi allo standard SAML v2.0 (SAML-Metadata), che prevede:
- l'elemento **<`EntityDescriptor`>** che contiene gli attributi:
  - **`entityID`**: indicante l'identificativo univoco (un URI) dell'entità;
  - **`ID`** univoco, per esempio basato su un Universally Unique Identifier (UUID) o su una combinazione origine + timestamp (quest'ultimo generato con una precisione di almeno un millesimo di secondo per garantire l'univocità);
- l'elemento **<`KeyDescriptor`>** contenente il certificato della chiave pubblica dell'entità, utile per la verifica della firma dei messaggi prodotti da tale entità nelle sue interazioni con le altre (SAML-Metadata);
- l'elemento **<`Signature`>** riportante la firma sui metadata. La firma deve essere prodotta secondo il profilo specificato per SAML (SAML-Metadata) utilizzando chiavi RSA almeno a 1024 bit e algoritmo di digest SHA-256 o superiore.

Dopo aver descritto il protocollo usato in SPID, vediamo come deve essere modificato per poter realizzare un sistema di firma elettronica avanzata.

3 Firma basata su SPID

La firma elettronica è un insieme dei dati in forma elettronica utilizzati come metodo di identificazione informatica. Essa è la forma più debole di firma in ambito informatico, in quanto non prevede meccanismi di autenticazione del firmatario o di integrità del dato firmato. Le organizzazioni internazionali di standardizzazione hanno definito molti formati di firme elettroniche che includono forme base di firma elettronica (ES-BES) e firme elettroniche avanzate (AdES) [6, 18].

In questa sezione descriviamo la proposta di utilizzo di una firma elettronica avanzata basata su SPID, la cui generazione consiste di due fasi, trattate nelle seguenti sottosezioni. Si consideri data un documento al quale apporre la propria firma digitale legata all'identità della persona che lo ha creato.

3.1 Generazione dell'impronta digitale

Nella prima fase viene applicata al documento in chiaro una funzione hash [2] che produce una stringa binaria di lunghezza costante, normalmente 160 o 256 bit, chiamata *digest*, ossia impronta digitale, che è una rappresentazione unica e compatta delle informazioni originali contenute nel documento. La funzione hash che si utilizza deve avere due proprietà fondamentali:
• unidirezionalità, ossia dato x è facile calcolare f(x), ma data f(x) è computazionalmente difficile risalire a x.
• priva di collisioni (collision-free), ossia deve essere computazionalmente impossibile trovare due messaggi a cui corrisponde la medesima impronta.
L'uso della funzione hash consente di evitare che per la generazione della firma sia necessario applicare l'algoritmo di cifratura, che è intrinsecamente inefficiente, all'intero testo che può essere molto lungo. Poiché la dimensione del *digest message* è, quasi sempre, molto più piccola di quella del messaggio originale, la generazione della firma risulta estremamente rapida.

3.2 Generazione della firma

Un'operazione preliminare che va fatta una tantum è la generazione delle chiavi. Mediante un software adatto al sistema crittografico adottato, si genera una coppia di chiavi da utilizzare: una, che verrà mantenuta segreta, per l'apposizione della firma; l'altra, che verrà resa pubblica, destinata alla verifica.
Una volta che le chiavi crittografiche sono disponibili, si può passare alla seconda fase, che è quella della vera e propria generazione della firma: essa consiste nella cifratura con la propria chiave privata dell'impronta digitale generata in precedenza. In questo modo la firma risulta legata, da un lato (attraverso la chiave privata usata per la generazione) al soggetto sottoscrittore, e dall'altro (per il tramite dell'impronta) al testo sottoscritto.
In realtà l'operazione di cifratura viene effettuata, anziché sulla sola impronta, su una struttura di dati che contiene altre informazioni utili, quali ad esempio l'indicazione della funzione hash usata per la sua generazione. Sebbene tali informazioni possano essere fornite separatamente rispetto alla firma, la loro inclusione nell'operazione di codifica ne garantisce l'autenticità.

3.3 Modifica del protocollo

In questa sezione descriviamo come implementare una firma elettronica avanzata attraverso SPID. In questo caso il ruolo di *Service Provider* è attuato da una parte che chiamiamo *Signature Provider*.
Per generare la firma di un documento, l'utente si connette al sito Web del Fornitore Firma (*Signature Provider*) e invia il documento da firmare insieme al proprio codice

di identificazione SPID. Quindi, il *Provider di Firma* calcola il *digest* utilizzando la funzione hash crittografica SHA-256.
Una volta creato il digest, viene creato l'`AuthnRequest`. La struttura di questo messaggio è la stessa descritta nella Definizione 1: tutte le informazioni sono le stesse di una richiesta SPID standard, eccetto il valore dell'attributo ID, che è il digest del documento (ricordiamo che nel protocollo SPID standard, ID è tipicamente il timestamp della richiesta).
In base all'XML Schema [7], l'ID deve essere un `NCName`, per cui deve rispettare le seguenti proprietà:

- Deve iniziare con una lettera o un carattere di sottolineatura o due punti
- Può contenere solo lettere, cifre, caratteri di sottolineatura, trattini e punti.
- I due punti devono essere utilizzati solo per separare i prefissi dei namespace dai nomi locali.

Inoltre, l'ID xsd comporta numerosi vincoli aggiuntivi:

- i loro valori devono essere univoci all'interno di un'istanza XML, indipendentemente dal nome dell'attributo o dal suo nome di elemento;
- un tipo complesso non può includere più di un attributo di tipo `xsd`: ID o qualsiasi tipo derivato da `xsd`: ID;
- gli attributi ID non possono avere valori predefiniti o fissi specificati.

A causa di queste limitazioni, il digest del documento non può essere direttamente utilizzato come ID ma è necessario operare una trasformazione che deve essere reversibile per renderlo compatibile con un tipo `NCName`.
La trasformazione che proponiamo di utilizzare per ottenere tale compatibilità è di far precedere il digest, rappresentato in codifica esadecimale, dal simbolo underscore "_", che è uno dei caratteri ammessi come iniziale. A questo punto, l'ID così ottenuto è un `NCName`, in quanto inizia per "_" ed è composto solo da cifre e caratteri. Ovviamente, in fase di verifica, per ottenere il digest a partire dall'ID, sarà sufficiente rimuovere il primo carattere dell'ID, cioè l'underscore.
Il messaggio dell'`AuthnRequest` viene rinviato all'utente per essere inoltrato all'Identity Provider. Se `AuthnRequest` è valido, l'Identity Provider esegue un'autenticazione con l'utente come mostrato negli **Step 3** e **4** del sistema SPID e, in caso di esito positivo, l'Identity Provider prepara l'Assertion.
In particolare, l'attributo `InResponseTo` contenuto nell'elemento <**`Subject`**> è impostato al valore dell'ID dell'`AuthnRequest`, che corrisponde al digest del documento da firmare. In questo modo, si stabilisce un collegamento tra l'*Assertion* e il corrispondente `AuthnRequest`. Nel nostro caso, l'Assertion rappresenta la firma del documento creato dall'*Identity Provider*. A questo punto, l'*Identity Provider* restituisce il messaggio `Response` contenente l'*Assertion*, che viene inoltrato al Signature Provider (**Step 5** e **6**). Una volta completata la generazione della firma, il *Signature Provider* crea la busta crittografica, che contiene le informazioni necessarie per la verifica della firma e abilita la possibilità di distribuire i documenti firmati.
Avendo modificato il meccanismo di firma, è necessario definire la struttura della busta crittografica da utilizzare: nella nostra proposta, la struttura che proponiamo di utilizzare per la busta crittografica assomiglia alla struttura di PKCS # 7 [17] ed è così definita:

Definizione 4. La busta crittografica firmata dall'*Identity Provider* contiene i seguenti campi:

- un attributo **`Version`**, che fornisce un numero di versione della sintassi per compatibilità con le revisioni future del documento;
- un attributo **`Content`**, che è il documento firmato;
- un attributo **`DigestAlgorithms`**, che indica l'algoritmo del messaggio-digest in base al quale il contenuto è digerito per il firmatario;
- un elemento finale <**`SignatureAssertion`**>, che è composto da campi, i cui valori vengono estratti da Assertion:
    - un unico attributo **`Digest`**. Il suo valore è uguale all'attributo InResponseTo dell'Assertion, che corrisponde al digest del documento;
    - un attributo **`Timestamp`**. Il suo valore è uguale a attributo IssueInstant, che specifica l'istante in cui il documento è stato emesso;
    - un elemento <**`Owner`**>. Questo valore è preso da <Subject> e indica il firmatario;
    - un elemento <**`SPIDcode`**>. Questo valore viene estratto dall'elemento AttributeStatement, che contiene il codice di identificazione SPID dell'utente;
    - un elemento <**`Issuer`**>. Questo è lo stesso di <**`Issuer`**>, che specifica l'EntityID del Provider di Identità;
    - un elemento <**`SPIDsignature`**>. Questo è lo stesso di <**`Signature`**>, che è la firma del Identity Provider sul documento.

Una volta completato il processo di firma, il Signature Provider invia la busta crittografica così generata all'utente (**Step 7**).

4 Conclusioni

Il processo di dematerializzazione della pubblica amministrazione prevede, alla sua base, la presenza di robuste infrastrutture normative, organizzative e tecnologiche che permettano di operare sui documenti informatici, sulla loro trasmissione, conservazione, gestione, con le stesse garanzie di legge previste nel dominio dei documenti cartacei. Tra queste infrastrutture, le firme elettroniche hanno certamente un posto di rilievo, essendo la firma l'istituto fondamentale utilizzato per fornire genuinità ed autenticità ai documenti. Il contributo di questo lavoro è approfondire e dettagliare una recente proposta di ricerca, che definisce un protocollo di firma elettronica avanzata attraverso l'adozione del sistema SPID, il sistema di identità digitale pubblica. Il risultato raggiunto è particolarmente interessante perché la firma ottenuta è sicura, portabile, verificabile in maniera ubiquitaria. Inoltre, essa è interoperabile al livello europeo, essendo previsto dal regolamento europeo eIDAS, che i sistemi di identità digitale pubblica, debbano essere interoperabili tra gli stati membri. L'approccio è pertanto applicabile su larga scala e utile ad accelerare i processi di dematerializzazione e di semplificazione per cittadini e imprese.

Riferimenti

1. Security Assertion Markup Language (SAML) V2.0 Technical Overview, http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-tech-overview-2.0.html, 2008
2. Funzione crittografica di hash, https://it.wikipedia.org/wiki/Funzione_crittografica_di_hash, 2019
3. SAML authentication, https://docs.citrix.com/en-us/netscaler/12/aaa-tm/saml-authentication.html, 2019
4. SPID - Regole Tecniche, https://media.readthedocs.org/pdf/spid-regole-tecniche/latest/spid-regole-tecniche.pdf, 2019
5. Il Regolamento UE n° 910/2014 - eIDAS, https://www.agid.gov.it/it/piattaforme/eidas, 2019
6. Firma elettronica avanzata FEA con SPID, https://blog.sygest.it/category/firma-digitale/, 2017
7. XML schema, http://www.datypic.com/sc/xsd/ns-xsd.html
8. Security Assertion Markup Language (SAML), http://it.wikipedia.org/wiki/Security_Assertion_Markup_Language, 2019
9. INPS: "Firma Elettronica Avanzata (FEA) con SPID, così semplifichiamo l'APE", https://www.agendadigitale.eu/cittadinanza-digitale/inps-firma-elettronica-avanzata-fea-con-spid-cosi-semplifichiamo-lape/, 2018
10. SPID, https://www.spid.gov.it/, 2019
11. F. Buccafurri, G. Caminiti, and G. Lax. The Dalì Attack on Digital Signature.Journal of Information Assurance and Security, 3:185–194, 2008
12. F. Buccafurri, L. Fotia, and G. Lax. Social signature. Signing by tweeting. In Electronic Government and the Information Systems Perspective, pages 1–14. Springer, 2014
13. F. Buccafurri, L. Fotia, and G. Lax. Implementing advanced electronic signature by Public Digital Identity System (SPID), Proc. of International Conference EGOVIS2016:289-303, Springer
14. F. Buccafurri, L. Fotia, G. Lax, and R. Mammoliti. Enhancing public digital identity system (spid) to prevent information leakage. In Electronic Government and the Information Systems Perspective, pages 57–70. Springer, 2015
15. C. Cuijpers and J. Schroers. eIDAS as guideline for the development of a pan European eID framework in FutureID. Open Identity Summit 2014, 237:23–38, 2014
16. J. Dumortier and N. Vandezande. Critical Observations on the Proposed Regulation for Electronic Identification and Trust Services for Electronic Transactions in the Internal Market. ICRI Research Paper, 9, 2012
17. B. Kaliski. Pkcs# 7: Cryptographic message syntax version 1.5. 1998
18. D. Pinkas, N. Pope, and J. Ross. Cms advanced electronic signatures (cades), IETF Request for Comments, 5126, 2008
19. T. Rabin. Robust sharing of secrets when the dealer is honest or cheating. Journal of the ACM (JACM), 41(6):1089–1109, 1994