UNIVERSITA' POLITECNICA DELLE MARCHE
FACOLTA' DI INGEGNERIA

Corso di Laurea triennale in ingegneria elettronica

Crittografia post-quantum per applicazioni spaziali

Post-quantum cryptography for space applications

Relatore:
Prof. Franco Chiaraluce

Tesi di Laurea di:
Lillini Davide

Correlatore:
Prof. Marco Baldi

A.A. 2019/ 2020

INDICE

I. INRODUZIONE

Il presente lavoro nasce dal presupposto che negli ultimi anni si è sviluppata una notevole attività di ricerca sui computer quantistici, dispositivi che sfruttano i fenomeni della meccanica quantistica per risolvere problemi matematici difficili o intrattabili per i computer convenzionali.

Ciò implica la concreta possibilità di compromettere la riservatezza e l'integrità delle comunicazioni digitali su Internet e altrove. La crittografia post-quantum (chiamata anche crittografia quantica resistente) opera per sviluppare sistemi crittografici che siano sicuri contro computer sia quantistici che classici e in grado di interagire con i protocolli e le reti di comunicazione esistenti [1].

Più in particolare, inizialmente l'obiettivo che ci si è posti è quello di studiare soluzioni per rendere sicuri i protocolli di comunicazione spaziale attualmente in uso per missioni deep space nei confronti di possibili attacchi condotti utilizzando computer quantistici. Ciò può essere fatto sostituendo le primitive crittografiche vulnerabili contenute all'interno di questi protocolli con primitive post-quantum. L'attenzione sarà focalizzata sulla crittografia asimmetrica (o a chiave pubblica) visto che, come sarà specificato nel seguito, questi algoritmi sono maggiormente esposti agli attacchi da computer quantistici.

In prima battuta sono stati analizzati i documenti del Consultative Committee for Space Data Systems (CCSDS) dedicati alla sicurezza in ambito spaziale, cercando di capire quali siano effettivamente le minacce per un sistema di comunicazione di questo tipo e come tali minacce possano avere impatto sulle varie tipologie di missione, classificate in base ai requisiti di sicurezza necessari. Poi si è fatta una panoramica sulle tipologie di algoritmi crittografici ritenuti sicuri in questo ambito, analizzando sia la parte riguardante gli algoritmi a chiave simmetrica sia quelli a chiave pubblica.

Dallo studio di questi documenti è emerso che la crittografia asimmetrica (di maggior interesse, per i motivi spiegati) viene utilizzata in ambito spaziale, in particolare deep space, solo per specifiche applicazioni, per lo più limitate all'autenticazione.

Per potere ampliare l'analisi, si è allora passati dagli scenari deep space a quello dei satelliti geostazionari, in ambito SIN (space internet network) analizzando nel dettaglio l'algoritmo AnFRA finalizzato all'accesso in roaming veloce e sicuro al SIN.

Da tale studio è emerso che all'interno del protocollo viene fatto uso dell'Elliptic Curve Digital Signature Algorithm (ECDSA) che, come noto in letteratura, risulta vulnerabile ad attacchi quantistici. Si è allora approfondita l'analisi dell'ECDSA, così da comprenderne meglio il funzionamento e cercare di proporre un sostituto post-quantum efficiente.

Dallo studio di ECDSA è emerso che il concetto matematico su cui si basa (curve ellittiche) è già stato preso in considerazione per proporre primitive post-quantum sicure, sfruttando un gruppo particolare di curve ellittiche chiamate 'supersingolari', le quali pure, alla fine della tesi, sono state

descritte, pur in modo introduttivo. Il presente lavoro potrà essere sviluppato ulteriormente in futuro attraverso la definizione di soluzioni alternative capaci di resistere alla sfida posta dai computer quantistici, anche in un ambiente complesso quale quello delle comunicazioni spaziali.

II. STANDARD CCSDS

Le tecnologie dell'informazione e della comunicazione sono progredite rapidamente negli ultimi anni e la connettività è diventata onnipresente. Questo vale anche per le infrastrutture delle missioni spaziali e pertanto minaccia la sicurezza dei nuovi campi applicativi. Allo stesso tempo, le missioni spaziali civili sono diventate parte di infrastrutture critiche come la navigazione, il meteo e le attività di risposta alle catastrofi. Di conseguenza, si devono prendere in considerazione meccanismi o impiegare politiche per mitigare tali rischi.

Nella scelta dei servizi di sicurezza appropriati per una determinata missione, il primo compito è quello di valutare le possibili minacce alla sicurezza del sistema. Questo compito fa normalmente parte dello sviluppo di una politica di sicurezza del sistema (SSP). La SSP è di regola un documento che fornisce una descrizione del sistema, gli obiettivi di sicurezza di primo livello e l'identificazione delle specifiche minacce alla sicurezza delle informazioni e alla disponibilità del sistema. Nel piano di sicurezza possono essere incluse altre informazioni, quali le norme di sicurezza e i requisiti di valutazione da applicare al sistema, le regole per l'accesso, il funzionamento dei sistemi critici per la sicurezza (ad esempio, l'autorità di certificazione) e, se viene utilizzata la cifratura, i mezzi con cui le chiavi sono distribuite e gestite.

La valutazione delle minacce dovrebbe constatare le vulnerabilità del sistema e quindi stabilire la probabilità, le conseguenze e il costo necessario a realizzare ciascun attacco all'infrastruttura. Una volta completato il processo di valutazione, è possibile identificare specifici servizi di sicurezza per contrastare i possibili rischi. La selezione delle contromisure richiede un'analisi costi-benefici per giustificare la spesa di implementazione dei servizi di sicurezza all'interno del sistema. Qualsiasi vulnerabilità ancora in essere è considerata "rischio residuo" e deve essere accettabile prima di mettere il sistema in produzione.

Per rendere più completa la trattazione dell'argomento, si fornisce di seguito una serie di informazioni più specifiche. Una minaccia può essere definita come una potenziale violazione della sicurezza. Tuttavia, rimane solo 'potenziale' fino a quando non si può dimostrare che esiste un'alta probabilità che la minaccia possa causare danni. In quel momento, essa viene ricategorizzata come una vulnerabilità.

Le minacce a un sistema di dati delle missioni spaziali includono:
a) distruzione non autorizzata di informazioni e/o risorse (ad esempio, veicoli spaziali o sistemi di terra);
b) corruzione o modifica non autorizzata di informazioni all'interno del sistema;
c) furto o perdita di informazioni e/o risorse;
d) divulgazione di informazioni a soggetti non autorizzati;
e) interruzione dei servizi.
Le minacce possono essere poi classificate come accidentali o intenzionali e possono essere attive o passive. Le minacce accidentali non hanno un intento premeditato e comprendono malfunzionamenti del sistema ed errori operativi; quelle intenzionali vanno dall'esame casuale delle informazioni del sistema ad attacchi sofisticati che utilizzano conoscenze specifiche dello stesso.
Molteplici servizi di sicurezza possono essere applicati a diversi livelli del sistema di comunicazione della navicella spaziale, e vengono scelti in base a diversi fattori:

a) la politica di sicurezza dell'Agenzia spaziale cui la missione fa capo;
b) i requisiti di sicurezza della missione;
c) i requisiti operativi della missione (compresi il supporto incrociato e l'interoperabilità);
d) lo standard raccomandato da CCSDS in uso;
e) la capacità dei sistemi di bordo.

Per definire una serie di requisiti di sicurezza generici per i diversi tipi di missioni spaziali, queste ultime sono state classificate come missioni che richiedono livelli di sicurezza elevati, moderati o minimi.

Missioni ad elevata sicurezza

Le missioni che richiedono un'elevata sicurezza sono generalmente associate al settore governativo o militare. L'accesso sicuro al sistema di controllo del veicolo spaziale è richiesto in ogni momento e in tutte le possibili condizioni operative o ambientali. Il database della missione deve essere protetto da accessi non autorizzati e devono essere attuate misure per impedire il rilevamento, l'intercettazione e l’utilizzo dei collegamenti di dati.

Per questa situazione sono previsti i seguenti requisiti:

a) protezione di tutti i dati di telecomando

- riservatezza,
- autenticazione,
- controlli di accesso,
- integrità dei dati (comprese le misure anti-replay),
- disponibilità;

b) protezione di tutti i dati telemetrici

- riservatezza,
- integrità dei dati,
- eventualmente altri servizi di sicurezza come l'autenticazione e il controllo degli accessi,
- disponibilità;

c) protezione di tutti i dati nel sistema di dati a terra

- riservatezza,
- autenticazione,
- integrità dei dati,
- disponibilità,
- controlli di accesso.

Missioni a sicurezza moderata

Le missioni che richiedono una sicurezza moderata possono includere missioni di carattere commerciale, meteorologico e di telerilevamento.

Queste ultime richiederanno la protezione dei veicoli spaziali e dei sistemi di terra da accessi non autorizzati e potrebbero dover proteggere i dati che sono commercialmente o operativamente sensibili o critici per la sicurezza. La protezione da accessi non autorizzati è particolarmente importante se la missione utilizza reti terrestri aperte (Internet) per fornire la connettività della stazione di terra.

Come minimo, le missioni con sicurezza moderata devono avere i seguenti requisiti:

a) protezione dei dati di telecomando

- autenticazione,
- integrità dei dati,
- possibile requisito di riservatezza;

b) protezione di alcuni o di tutti i dati telemetrici
- riservatezza,
- integrità dei dati;
c) protezione di alcuni o di tutti i dati nel database a terra
- autenticazione,
- integrità dei dati,
- possibile requisito per il controllo degli accessi,
- possibile requisito di riservatezza.

Missioni a sicurezza minima

Le missioni che richiedono una sicurezza minima includono tutte le altre missioni spaziali. È probabile che queste missioni richiedano la sicurezza del sistema di telecomando per impedire l'accesso non autorizzato o la manomissione dei dati, sia intenzionale che non intenzionale. Ci possono essere anche requisiti di riservatezza per specifiche informazioni di telemetria (ad esempio, dati scientifici proprietari, immagini).
Le missioni con sicurezza minima devono avere i seguenti requisiti:
a) protezione di tutti i dati di telecomando
- autenticazione,
- integrità dei dati,
- possibili requisiti di riservatezza;
b) protezione di alcuni dati telemetrici: riservatezza, integrità dei dati;
c) protezione di alcuni dati del database a terra: riservatezza, integrità dei dati, controllo degli accessi.

Informazioni più dettagliate sono disponibili in [2], [3].

I requisiti di sicurezza per le varie categorie di missioni citate sopra possono essere implementati attraverso l'utilizzo di algoritmi specifici raccomandati dal CCSDS [4], [5].
Di seguito introduciamo alcuni elementi esplicativi dei concetti di autenticazione e integrità.
L'autenticazione può essere utilizzata per identificare in modo univoco una persona o un'entità. Può anche essere usata per indicare un "ruolo" che un individuo ha assunto (ad esempio, il controllore dello strumento X). Oppure, può essere applicata per identificare in modo univoco una stazione di lavoro o un gruppo di stazioni di lavoro che compongono un centro di controllo. In questo modo, qualsiasi informazione ricevuta che si ritiene sia stata inviata da un soggetto che ricopre un

determinato ruolo (ad esempio, controller dello strumento X), o da una struttura (ad esempio, il centro di controllo della missione) può essere autenticata come effettivamente inviata dall'identità rivendicata. Il destinatario ha la garanzia che l'origine della fonte dei dati è autentica (ad esempio, persona, luogo, ruolo) e che i dati stessi non sono stati alterati o modificati in transito senza autorizzazione o notifica.

Per gli ambienti che utilizzano chiavi simmetriche (potenzialmente insieme alla cifratura simmetrica), si deve utilizzare uno dei due tipi di algoritmi per fornire l'autenticazione / integrità: basato su hash o su cifratura [5].

Gli algoritmi di autenticazione del messaggio (MAC) basati sull'hash utilizzano funzioni di hash crittografiche (ad es. SHA-256, dove SHA è l'acronimo di Secure Hash Algorithm) ed una chiave. I dati da autenticare vengono concatenati con la chiave condivisa e poi l'algoritmo di hash viene eseguito sui dati concatenati, ottenendo un MAC di dimensioni fisse. La dimensione del digest del messaggio dipende strettamente dall'algoritmo di hash utilizzato.

Al posto di un MAC basato sull'hash si può costruire un MAC basato su un cifrario. Il MAC basato sulla cifratura utilizza un algoritmo crittografico (ad es. l'Advanced Encryption Standard (AES)). Il 'segreto' condiviso viene utilizzato come chiave crittografica per l'algoritmo che fornisce un MAC come risultato.

I MAC basati sulla cifratura possono utilizzare meglio le risorse disponibili in quanto sono necessari sia per l'autenticazione che per la riservatezza, e quindi un unico algoritmo può essere utilizzato per entrambi. Inoltre, i MAC basati sulla cifratura possono essere implementati più facilmente in hardware rispetto ai MAC basati sull'hash.

L'utilizzo di algoritmi simmetrici viene preferito in ambito spaziale rispetto a quelli asimmetrici in quanto hanno la possibilità di precaricare le chiavi a bordo dei dispositivi. Così facendo, si alleggerisce di molto il carico computazionale che il dispositivo deve compiere per poter convalidare i messaggi.

Per gli ambienti in cui è disponibile la crittografia a chiave pubblica, l'autenticazione e l'integrità possono essere realizzate utilizzando un algoritmo di firma digitale.

Una possibile soluzione è la seguente. Il 'firmatario' esegue un hash sui dati da firmare utilizzando un algoritmo di hash (ad esempio, SHA). La parola hash risultante viene quindi criptata utilizzando la chiave privata del firmatario per creare la firma digitale.

Il destinatario dei dati firmati verifica la firma sui dati ricevuti per assicurare che essi provengano dall'entità rivendicata e non siano stati modificati. Per autenticare la firma, il digest del messaggio viene decifrato utilizzando la chiave pubblica del firmatario, che può essere inviata con i dati (e autenticata separatamente tramite la firma di un'autorità di certificazione). Se ottenuta in precedenza,

questa potrebbe già essere memorizzata nella cache dal destinatario. Oppure può essere ottenuta da un server di chiavi pubbliche. Se la decrittazione del messaggio digest è riuscita, ciò dimostra l'autenticità dell'identità del firmatario. L'algoritmo di hash viene quindi eseguito sui dati ricevuti e la parola hash risultante viene confrontata con la parola hash trasmessa e decifrata. Se sono identici, l'integrità dei dati è garantita. Ciò dimostra che non si è verificata alcuna modifica non autorizzata o accidentale dei dati durante la trasmissione e che i dati ricevuti sono esattamente gli stessi dati trasmessi dalla fonte.

Se invece si vuole fornire simultaneamente riservatezza, integrità e autenticità si deve fare ricorso alla crittografia autenticata. In generale la cifratura autenticata può essere eseguita combinando un algoritmo di cifratura con un algoritmo di autenticazione (ad es. MAC), a condizione che entrambi siano noti per essere sicuri contro gli attacchi.

Gli algoritmi di cifratura autenticata consigliati dal CCSDS sono sostanzialmente due: il Counter Mode con CBC-MAC (CCM) [4] e il Galois/Counter Mode (GCM)[5].

Lo standard CCSDS fornisce dettagli anche su la parte di gestione delle chiavi [6] e consiglia i migliori protocolli di sicurezza andando ad analizzare anche tutte le procedure di implementazione [7].

Sostanzialmente i documenti CCSDS forniscono una linea guida che consente di capire quali sono le minacce da cui difendersi e soprattutto come implementare protocolli sicuri attraverso l'applicazione di algoritmi standardizzati.

III. AUTENTICAZIONE IN AMBITO SPAZIALE E PROBLEMI APERTI

L'autenticazione dei dati si ottiene aggiungendo un'unità di informazione supplementare al messaggio originale. Le informazioni supplementari si presentano sotto forma di firma digitale o di digest del messaggio. Come illustrato nella figura 1.

La firma digitale identifica in modo univoco l'origine dei dati in modo tale che il destinatario abbia la certezza che le informazioni provengano dalla fonte rivendicata. La caratteristica essenziale di tale meccanismo è che i dati firmati non possono essere creati da un'entità non autorizzata.

Figura 1.

Molti meccanismi di generazione della firma digitale richiedono l'uso di un algoritmo crittografico asimmetrico per il quale mittente e ricevente non hanno le stesse chiavi crittografiche. Piuttosto, vengono utilizzate una coppia di chiavi pubbliche e private matematicamente correlate tra loro. All'origine dei dati, l'algoritmo crittografico genera una firma digitale utilizzando la chiave privata del mittente. La firma può essere generata dai dati stessi ed è di una lunghezza specifica, a seconda dell'algoritmo utilizzato. L'autenticazione dell'origine dei dati si ottiene quando la firma digitale viene verificata con successo dal destinatario utilizzando la chiave pubblica del mittente.

Per l'autenticazione può essere utilizzato anche un message digest, che genera un valore di controllo univoco indicando al destinatario che i dati non sono stati modificati. Il digest dei messaggi viene generato su un 'segreto' condiviso che solo il mittente e il destinatario possiedono.

La crittografia dei dati stessi può anche fornire un'autenticazione implicita quando si utilizza un algoritmo crittografico simmetrico. L'autenticazione si ottiene in quanto il destinatario deve avere e utilizzare la chiave corretta per decifrare la firma digitale allegata ai dati. Ciò presuppone l'esistenza di un meccanismo di distribuzione della chiave sicuro. Anche la crittografia asimmetrica (chiave pubblica) può fornire un'autenticazione nel momento in cui c'è la garanzia che la chiave pubblica è legata all'originatore (ad esempio, firmata da un'autorità di certificazione). Tuttavia, è necessario usare cautela perché l'autenticazione può essere compromessa se i dati crittografati vengono intercettati e successivamente riprodotti.

Dallo studio del materiale fornito dal CCSDS non si è riscontrato nessun dato sensibilmente utile, in quanto la trattazione è puramente incentrata su quali procedure seguire per rendere sicuro un sistema di comunicazione. Non sono inoltre presenti riferimenti espliciti su protocolli già utilizzati, pertanto è stato necessario ampliare la sfera di ricerca tentando di trovare del materiale su cui lavorare, ma nemmeno questa strada ha portato i frutti attesi.

Sono così giunto alla conclusione che in ambito spaziale, e in particolare in applicazioni in deep space, la crittografia asimmetrica viene utilizzata in rari casi solo ai fini dell'autenticazione, dato che è caratterizzata da una notevole complessità di calcolo e scambio delle chiavi. Risulta quindi difficile da implementare su velivoli spaziali dotati di una limitata capacità di calcolo, per questo si predilige la crittografia simmetrica che ha un costo computazionale minore e una ridotta dimensione delle chiavi.

IV. AnFRA

Rispetto ai tradizionali sistemi di comunicazione senza fili, come le reti cellulari, il sistema di comunicazione satellitare ha caratteristiche di copertura globale, grande efficienza, flessibilità della larghezza di banda e non è limitato da alcuna condizione geografica tra due punti di comunicazione. Analogamente, anche il servizio di roaming deve essere fornito da SIN, cosicché gli utenti delle reti wireless tradizionali possano accedervi per ottenere servizi di rete, soprattutto in alcune condizioni estreme, come in mare, nel deserto o in zone colpite da terremoti, dove non esiste una stazione base per l'accesso degli utenti alle reti wireless tradizionali.

Per la sicurezza e la qualità del servizio di roaming è fondamentale per SIN implementare un sistema di autenticazione sicura. Nelle reti wireless tradizionali il protocollo di autenticazione in roaming può essere classificato in accordo con due distinte tipologie: schema di autenticazione in roaming a tre parti e schema di autenticazione in roaming a due parti. Gli schemi di autenticazione in roaming a tre parti, come [8] e [9], di solito verificano l'utente in roaming sul suo server di casa, in modo che il server straniero non possa violare la privacy degli utenti. Tuttavia, essi necessitano di maggiori interazioni e non possono essere implementati nell'architettura SIN, poiché il SIN ha un lungo ritardo di propagazione tra i satelliti e la terra. Anche per i satelliti con orbita terrestre bassa (LEO), che sono più vicini al suolo, ci sono ancora tra i 500 e 2.000 chilometri di distanza, e un conseguente ritardo di propagazione che può variare dai 10 ai 40 ms. Ciò causerà un intollerabile ritardo di autenticazione a questi schemi.

Gli schemi di autenticazione in roaming a due parti, d'altro canto, autenticano gli utenti in roaming senza richiedere la partecipazione del proprio server domestico e di solito richiedono meno interazioni, il che può ridurre notevolmente il ritardo di autenticazione. Tuttavia, i suddetti schemi esistenti non possono ancora essere implementati direttamente nel SIN, dal momento che solitamente sono caratterizzati da alcune operazioni che richiedono molto tempo per essere eseguite, come il controllo delle liste di revoca.

Nel frattempo, il lungo ritardo di propagazione non può essere ridotto in modo significativo, poiché in questi schemi esistono ancora interazioni multiple tra satelliti e dispositivi a terra.

Con il recente sviluppo della tecnologia hardware satellitare, ora i satelliti sono in grado di gestire calcoli più complessi. Dato questo upgrade, ora i satelliti possono essere utilizzati come verificatori al posto dei server di terra, il che riduce ampiamente le interazioni tra le due parti e riduce il ritardo di autenticazione. Tuttavia, oltre al problema del lungo ritardo di propagazione, anche i requisiti di sicurezza per lo scenario di roaming in SIN sono difficili da garantire.

In primo luogo, a causa della vulnerabilità del SIN, alcuni attacchi malevoli come gli attacchi di intercettazione, modifica, replay e impersonificazione possono facilmente danneggiare il sistema. In secondo luogo, i link altamente esposti del SIN potrebbero essere utilizzati dagli attaccanti per compromettere la privacy degli utenti attraverso l'intercettazione di un canale scoperto. Infine, anche le entità estere della rete potrebbero essere potenziali avversari, poiché potrebbero facilmente rivelare la privacy degli utenti rintracciandone identità e ubicazione.
Nel documento [10], viene proposto uno schema di autenticazione basato sulla firma di gruppo per proteggere la privacy e fornire un accesso rapido per gli utenti in roaming. Nello schema in esame ogni LEO funge da verificatore per l'autenticazione degli utenti mobili quando richiedono di accedere al SIN, il che può ridurre ampiamente il ritardo di autenticazione e il numero di interazioni. Inoltre l'utilizzo della firma di gruppo può fornire in modo efficiente l'anonimato degli utenti, in modo che la privacy non venga divulgata.
In particolare lo schema proposto in [10] fornisce i seguenti vantaggi:

1) rafforzamento della funzione di autenticazione dei satelliti LEO;
2) fornitura di uno schema di autenticazione rapida in roaming, denominato AnFRA;
3) ottenimento di un'autenticazione di accesso rapida tra utenti e satelliti;
4) implementazione di un meccanismo di pre-negoziazione per accelerare l'autenticazione;
5) autenticazione degli utenti senza la partecipazione del server domestico;
6) garanzia di anonimato e di sicurezza agli utenti;
7) meccanismo ben progettato per supportare la revoca dinamica dell'utente, con conseguente risparmio di tempo per l'implementazione della lista di revoca al momento dell'autenticazione.

Per l’articolo completo e dettagli più specifici si rimanda a [10].

Andando ad analizzare l’articolo nel dettaglio, si è riscontrato che il punto debole di questo protocollo sta nell’utilizzo dell’algoritmo ECDSA, che sappiamo essere quantum-vulnerabile e quindi deve essere sostituito. In AnFRA ECDSA viene utilizzato esclusivamente per garantire l’integrità dei parametri nelle varie procedure e per verificare le identità dei vari enti in gioco, più precisamente si utilizza l’ECDSA per generare le chiavi di verifica/firma, per firmare i messaggi e per verificare le firme. In questa specifica applicazione si è scelto di utilizzare questo algoritmo: essendo basato sulle curve ellittiche, esso consente di ottenere chiavi di ridotte dimensioni, pur garantendo comunque uno standard molto elevato riguardo i servizi di autenticazione, integrità e riservatezza. Questa sua caratteristica risulta molto utile, in quanto dovrà essere implementato su dispositivi con capacità di calcolo limitate.

Le caratteristiche principali di un sostituto post-quantum dell'ECDSA devono comunque essere:

- resistenza ad attacchi quantistici;
- dimensione della chiave contenuta garantendo comunque una sicurezza elevata;
- capacità di generare chiavi di verifica/firma, firmare i messaggi e verificare le firme;
- basso costo temporale per l'esecuzione delle funzionalità al punto precedente, così da non avere ritardi significativi nell'autenticazione.

Nella prossima sezione andremo a proporre degli approfondimenti sulle curve ellittiche e in particolare su ECDSA cosi da poterne comprendere meglio il funzionamento e discutere possibili soluzioni post-quantum.

V. ECDSA

Quando le informazioni vengono trasmesse da una parte all'altra, il destinatario può desiderare di sapere che i dati non siano stati modificati durante il transito. Inoltre, potrebbe voler essere certo dell'identità da cui proviene l'informazione. L'uso di firme digitali con crittografia a chiave pubblica può fornire la garanzia: (1) dell'identità del firmatario e (2) che il messaggio ricevuto non sia stato modificato durante la trasmissione.

Una firma digitale è un analogo elettronico a una firma scritta. Può essere utilizzata per dimostrare a terzi che le informazioni sono state, di fatto, firmate dal mittente rivendicato. A differenza delle loro controparti scritte, le firme digitali verificano anche l'integrità delle informazioni.

L'ECDSA viene utilizzato da un mittente per generare una firma digitale su dei dati e da un destinatario per verificare l'autenticità della firma. Ogni utente ha una chiave pubblica e privata: la chiave privata viene utilizzata nel processo di generazione della firma e la chiave pubblica viene utilizzata nel processo di verifica della firma. Sia per la generazione che per la verifica, il messaggio k viene compresso tramite l'algoritmo SHA-256, prima della generazione della firma e del processo di verifica.

Un avversario, che non conosce la chiave privata del firmatario, non può generare una copia della firma. In altre parole, le firme non possono essere falsificate. Tuttavia, utilizzando la chiave pubblica del firmatario, chiunque può verificare un messaggio validamente firmato.

L'utente di una coppia di chiavi privata/pubblica richiede la garanzia che la chiave pubblica rappresenti il proprietario di tale coppia di chiavi. In altre parole, deve esserci un legame tra l'identità di un proprietario e la chiave pubblica d. Questa relazione può essere certificato da un ente reciprocamente attendibile. Ciò può essere ottenuto utilizzando un'autorità di certificazione che genera un certificato.

Questo standard consente di rilevare messaggi duplicati e impedire l'accettazione di messaggi corrotti quando il messaggio firmato include:

1. l'identità del destinatario previsto;
2. un identificatore di messaggio (MID).

Il firmatario fornisce e mantiene il controllo di tutto il materiale di codifica. Nel sistema crittografico asimmetrico ECDSA, l'integrità dei dati firmati dipende da:

1. la prevenzione della divulgazione, dell'uso, della modifica, della sostituzione, dell'inserimento e dell'eliminazione non autorizzati della chiave privata, *d*, del valore del messaggio k;
2. la prevenzione della modifica, sostituzione, inserimento e eliminazione non autorizzati dei parametri del dominio della curva ellittica per le procedure di calcolo ECDSA.

Pertanto, *se d* viene divulgato, l'integrità di qualsiasi messaggio firmato utilizzando tale *d* non può più essere garantita. Analogamente, i valori per i parametri del dominio della curva ellittica devono essere protetti.

La generazione di chiavi deve essere eseguita su apparecchiature fisicamente isolate in modo che, in caso di guasto hardware o software, non venga conservata alcuna informazione parziale. Ad esempio, se un arresto anomalo del sistema causa un core dump, alcuni dei dati del materiale di codifica possono essere recuperati.

Prima di introdurre in maniera più dettagliata ECDSA verranno riportate delle nozioni base riguardanti la teoria dei campi, il problema del logaritmo discreto e le curve ellittiche.

Definizione 1. Un *gruppo è* una struttura algebrica composta da:

- Un insieme di elementi G
- Operazione chiusa sull'insieme G, che è associativa. Cioè $(a \cdot b) \, c = a \cdot (b \cdot c)$, per $a,b,c \in G$
- Elemento identità.
- L'esistenza di inversi sotto l'operazione impostata

Un gruppo in cui l'operazione impostata è anche commutativa *(cioè* $a \cdot b = b \cdot a$) è noto come gruppo *abeliano.*

Ad esempio, si potrebbe usare + come operazione e 0 come elemento identità. Utilizzando questi e l'insieme di interi Z, otteniamo un gruppo valido, in cui usiamo interi di segni opposti come coppie di inversi. Al contrario, l'insieme dei numeri naturali N non forma un gruppo in quanto non si possono definire gli inversi.

Definizione 2. Sia a un elemento del gruppo G con *identità* 1 e $\cdot$ come operazione di gruppo. L'ordine di a, è l'intero più piccolo n tale che:

$$\underbrace{a \cdot a \cdot a \cdot \ldots \cdot a}_{n} = 1$$

L'insieme $\{a,a^2,a^3,a^4,\ldots,a^n\}$ forma un sottogruppo ciclico di G di ordine n, dove a è chiamato 'generatore' per quel sottogruppo.

Definizione 3. Un *campo è* una struttura algebrica in cui:

- Abbiamo un insieme di elementi G chiusi sotto moltiplicazione e addizione.
- L'addizione e la moltiplicazione sono entrambe associative sotto l'insieme G.
- L'addizione e la moltiplicazione sono entrambe commutative sotto l'insieme G.
- Abbiamo l'esistenza di inversi sia sotto l'addizione che con la moltiplicazione.
- La moltiplicazione è distributiva rispetto all'addizione: $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$.
- Abbiamo identità distinte per la moltiplicazione e l'addizione.

Chiamiamo caratteristica di un campo F il più piccolo numero n tale che sommando n copie di 1 la caratteristica è uguale a 0. Ad esempio, se la caratteristica di F, $char(F)$, è 2 e l'identità in F è 1, allora $1 + 1 = 0$. Se $char(F) = 3$, allora $1 + 1 + 1 = 0$.

I campi Galois sono campi costituiti da un numero finito di elementi. Un esempio di tale campo sono i moduli dei numeri primi interi p, $\mathbb{Z}/\mathbb{Z}_p$. Con gli elementi p, da 0 a $p - 1$, denotiamo questo insieme come $GF(p)$. Si noti che questo campo è la combinazione di due gruppi abeliani, uno con addizione e identità 0 e uno con moltiplicazione e identità 1; in quest'ultimo caso 0 non è preso come parte del gruppo per preservare l'esistenza di inversi.
Ora denotiamo il problema del logaritmo discreto sul gruppo moltiplicativo degli interi modulo p, $\mathbb{Z}/\mathbb{Z}_p$, come segue. Dato $g,a \in \mathbb{Z}/\mathbb{Z}_p$, dove a è un membro del sottogruppo ciclico generato da g, trovare un intero k tale che:

$$g^k \equiv a \ mod \ p$$

Definizione 4. Denotiamo il problema del factoring discreto come segue. Dato un numero N, il fattore di due grandi primi p e q, trovare p e q.

Prima di procedere, è importante fornire alcune informazioni. La crittografia a curva ellittica fu introdotta nel 1985 da Victor Miller e Neal Koblitz che svilupparono entrambi indipendentemente l'idea di usare le curve ellittiche come base di un gruppo per il problema del logaritmo discreto. Si ritiene che esso garantisca più sicurezza rispetto ad altri gruppi e offra dimensioni di chiave molto più piccole. Di conseguenza le curve ellittiche hanno rapidamente guadagnato interesse.

Curve ellittiche

Definiamo una curva ellittica $E(F)$ come un insieme di punti in un campo F, soddisfacendo un'equazione della forma:

$$y^2 + a_1xy + a_2y = x^3 + a_3x^2 + a_4x + a_5$$

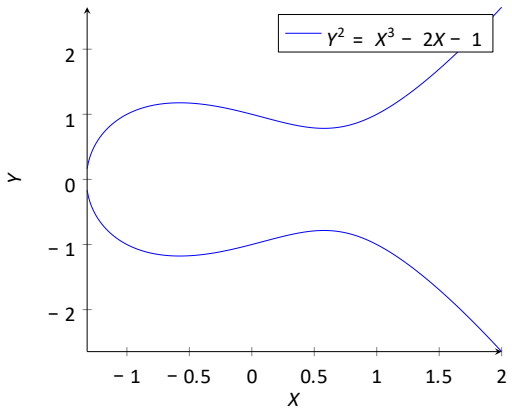
Dove $a_1, a_2, .., a_5 \in F$. Se assumiamo che la caratteristica del campo sia diversa da 2, allora questa equazione può essere semplificata in:

$$y^2 = x^3 + a_3x^2 + a_4x + a_5$$

Questa equazione può essere ulteriormente semplificata se la caratteristica del campo è anche diversa da 3, dando così l'equazione più familiare, nota come forma normale di Weierstrass:

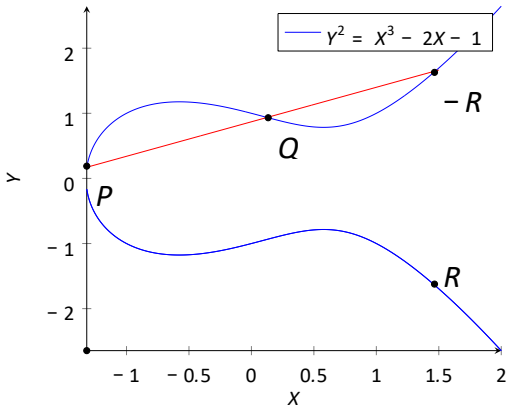
$$y^2 = x^3 + ax + b \mid a=a_4, b=a_5,$$

un esempio di curva ellittica in $\mathbb{R}^2$ può essere il seguente



Possiamo definire un gruppo G sull'insieme delle soluzioni dell'equazione della curva come segue. L'insieme dei punti sulla curva E sono gli elementi del gruppo G. Aggiungiamo un altro punto, il "punto all'infinito" (∞,∞), come identità, e lo chiamiamo O.

Si noti che per ogni *punto* $P = (x,y)$ sulla curva, anche il punto $P' = (x,-y)$ deve essere sulla curva. Lasciamo che P e P'siano inversi nel gruppo G. In particolare, $P + P' = O$ *se* '+' è la legge di gruppo. Tutto ciò di cui abbiamo bisogno ora è un'operazione chiusa '+' sul gruppo G. Dati i punti P e Q sulla curva E e i membri di G, sia $R = P + Q$, il punto tale che $P + Q - R = O$. Nel campo $\mathbb{R}^2$, c'è una facile costruzione geometrica per R che viene illustrata di seguito.



Si noti che tale costruzione è sempre possibile a meno che $P = Q$ o $Q = -P$ (cioè la linea che attraverso P e Q sia verticale). Nel primo caso, si prende semplicemente la tangente alla curva nel punto P e si utilizza la seconda intersezione con la curva come punto $-R$. Nel secondo caso, lasciamo $P + (-P) =$ O, e allo stesso modo $P + O = P$.

Se $P(x_1,y_1)$ e $Q(x_2,y_2)$ sono punti distinti di E, allora possiamo esprimere le coordinate del punto $R(x_3,y_3)$ risolvendo l'intersezione tra la linea che attraversa P e Q e la curva E:

$$\begin{cases} y = \dfrac{(y_2 - y_1) \cdot x + y_1 x_2 - y_2 x_1}{x_2 - x_1} \\ y^2 = x^3 + ax + b \end{cases}$$

Quindi le coordinate di $R(x_3,-y_3)$ sono date da:

$$x_3 = (\frac{y_2 - y_1}{x_2 - x_1})^2 - x_2 - x_1$$
$$y_3 = -\left(y_2 + \frac{y_2 - y_1}{x_2 - x_1}(x_3 - x_2\right)$$

Se $P = Q$, prendiamo l'intersezione tra la linea tangente e la curva E, che ci dà un'espressione più semplice per $R(x_2,y_2)$:

$$x_2 = \frac{3x_1^2 + a}{2y_1} - 2x_1$$

$$y_2 = -\left(y_1 + \frac{3x_1^2 + a}{2y_1}(x_2 - x_1\right)$$

Va notato che questa espressione include il parametro a della curva e non b poiché prendiamo la derivata della curva per ottenere la pendenza della linea tangente. Utilizzando questa operazione, si ricava la definizione di *moltiplicazione scalare* sul gruppo G, aggiungendo ripetutamente un punto a se stesso:

$$nP = \underbrace{P + P + P + \cdots + P}_{n}$$

Mentre l'elaborazione *di nP* può essere eseguita in modo efficiente usando l'algoritmo double and add [12], il problema opposto, vale a dire trovare n dato P *e* nP, può essere reso un problema particolarmente difficile quando la curva ellittica viene presa su un campo finito. Questo è noto come il problema del logaritmo discreto della curva ellittica, per il quale non si conosce alcun algoritmo efficiente. Inoltre, mentre il problema del factoring discreto può essere risolto in tempo sub esponenziale, i migliori algoritmi noti per risolvere il problema del logaritmo discreto della curva ellittica sono puramente esponenziali.

Algoritmo di firma digitale a curva ellittica (ECDSA). Supponiamo che Alice voglia firmare un messaggio *m* in modo che Bob sappia senza dubbio che il messaggio è stato inviato da Alice. Al fine di ottenere questo risultato si può utilizzare il protocollo ECDSA, che funziona come segue:

(1) Per alcuni messaggi *m*, Alice accetta un hash di *m* e lo tronca in modo che abbia lunghezza in bit pari a *k*, dove k è l'ordine del sottogruppo generato da *P* in *G*. Sia *z* l'hash troncato risultante di *m*.

(2) Alice sceglie un numero intero *n* tale che $1 \leq n \leq k$

(3) Alice calcola $Q = nP$ e *r*, la coordinata *x* del punto *Q*, modulo *k*. Se $r = 0$, tornare al passaggio 1.

(4) Alice calcola quindi $s = n^{-1}(z+rn_A)$, dove n_A è la chiave privata di Alice e n^{-1} è l'inverso moltiplicativo di *n* mod *k*. Se $s = 0$, tornare al passaggio 1.

(5) La coppia (*r,s*) forma la firma.

Si noti che il passaggio 4 richiede il calcolo dell'inverso moltiplicativo di *n* mod *k*. Tuttavia questo è possibile solo quando l'ordine del sottogruppo di *G* è esso stesso primo. Le curve con ordine primario sono quindi essenziali nell'uso di ECDSA. Ora, per verificare la firma di Alice, Bob utilizza la chiave pubblica *A* di Alice e procede come segue:

(1) Bob calcola $u_1 = s^{-1}z$ mod z

(2) Bob calcola $u_2 = s^{-1}r$ mod z

(3) Infine, Bob calcola il punto $Q = u_1P + u_2A$ e verifica $r = x_Q$ mod *n*

Come detto detto in precedenza il principale problema delle curve ellittiche per l'era post-quantum sta nel fatto che il problema del logaritmo discreto basato sulle curve ellittiche è risolvibile in tempo polinomiale utilizzando l'algoritmo di Shor [12].

Peter Shor infatti ha dimostrato, nel 1994, che utilizzando un computer quantistico, sarebbe stato possibile risolvere il problema del factoring discreto in tempo polinomiale, cosa che si ritiene impossibile sui computer classici. Nel suo articolo, Shor presentò due algoritmi simili, uno per il factoring discreto e l'altro per il problema del logaritmo discreto. Fu quindi mostrato che l'algoritmo di Shor poteva essere esteso al problema discreto del logaritmo su qualsiasi gruppo abeliano.

Ma dato che tale algoritmo è applicabile solo su curve ellittiche definite su gruppi abeliani, il problema si può aggirare utilizzando una diversa famiglia di curve ellittiche chiamate 'supersingolari'.

Quindi si assiste allo sviluppo di un nuovo metodo di crittografia, basato anche su curve ellittiche, che ha dato speranza di essere resistente ai quanti. Questo metodo si basa su isogenie di curve ellittiche supersingolari ed è stato suggerito da De Feo e al. A causa della sua recente scoperta, questo

schema di crittografia richiederà più attenzione e ricerca da parte della comunità di crittografia al fine di affermarne la sicurezza.

La ragione principale per cui si ritiene che lo schema sia resistente ai quanti è che si basa su un gruppo non abeliano: l'insieme delle isogenie di una data curva ellittica e l'operazione di composizione. Questo è fondamentalmente diverso da l'altro protocollo discusso in questo documento, che si basa su gruppi abeliani, e quindi è candidato all'attacco di Shor. Attualmente è una domanda aperta nel calcolo quantistico se lo stesso tipo di algoritmo a tempo polinomiale esista per gruppi non abeliani.

Per le sue piccole chiavi, il protocollo basato sull'isogenia ha argomenti forti per sostituire le attuali curve ellittiche in uso. Saranno tuttavia necessarie ulteriori ricerche prima che la loro sicurezza sia sostenuta fino all'uso pratico. Ci sono anche molti altri metodi classici a chiave pubblica che sono basati su gruppi non abeliani e si ritiene che siano resistenti ai quanti. Si può citare la crittografia su reticolo, che è stata maggiormente studiata fino ad oggi, o le equazioni multivariate, i codici di errore e la crittografia basata su hash.

Per informazioni più dettagliate su questo capitolo consultare i riferimenti [11] e [12]

VI. SCHEMI DSA POST-QUANTUM

Negli ultimi anni si è sviluppata una notevole quantità di ricerca in merito ai computer quantistici, macchine che sfruttano i fenomeni della meccanica quantistica per risolvere problemi matematici difficili o intrattabili per i computer convenzionali.

Ciò presuppone la seria possibilità di compromettere la riservatezza e l'integrità delle comunicazioni digitali su Internet e altrove. L'obiettivo della crittografia post-quantum (chiamata anche crittografia quantica resistente) è sviluppare sistemi crittografici che siano sicuri contro computer sia quantistici che classici e in grado di interagire con i protocolli e le reti di comunicazione esistenti.

Molti degli attuali protocolli di comunicazione più importanti si basano principalmente su tre funzionalità crittografiche fondamentali: crittografia a chiave pubblica, firme digitali e scambio di chiavi. Attualmente, queste funzionalità vengono implementate principalmente utilizzando lo scambio di chiavi Diffie-Hellman, il crittosistema RSA (Rivest-Shamir-Adleman) e i crittosistemi a curva ellittica. La sicurezza di questi dipende dalla difficoltà di alcuni problemi di teoria dei numeri come la fattorizzazione di numeri interi.

Nel 1994 si è dimostrato che i computer quantistici possono risolvere efficacemente ciascuno di questi problemi basati sulla teoria dei numeri, rendendo impotenti tutti i crittosistemi a chiave pubblica basati su tali presupposti.

Pertanto, indipendentemente dal fatto che possiamo stimare il tempo esatto dell'arrivo dell'era dell'informatica quantistica, dobbiamo iniziare ora a preparare i nostri sistemi di sicurezza delle informazioni per essere in grado di resistere alle capacità del calcolo quantistico.
Come accennato in precedenza, la costruzione di un computer quantistico su larga scala renderebbe insicuri molti degli attuali crittosistemi a chiave pubblica. In particolare, questo include quelli basati sulla difficoltà della fattorizzazione dei numeri interi, come RSA, nonché quelli basati sul problema del logaritmico discreto. Al contrario, l'impatto sui sistemi a chiavi simmetriche non sarà così drastico: sulla base di prime valutazioni, dovrebbe essere sufficiente raddoppiare la lunghezza della chiave per compensare l'effetto dell'algoritmo di Grover.
Tale algoritmo, infatti, fornisce una velocità quadratica per gli algoritmi di ricerca quantistica rispetto agli algoritmi di ricerca sui computer classici.
Di conseguenza, la ricerca di algoritmi ritenuti resistenti agli attacchi dei computer sia classici che quantistici si è concentrata su algoritmi a chiave pubblica. Ora verrà fornita brevemente una panoramica delle principali famiglie per le quali sono state proposte le primitive post-quantum.

Crittografia basata su reticoli (Lattice based)

Un reticolo è una griglia infinita di punti; il problema computazionale su cui si basa la tecnologia lattice-based è il "Shortest Vector Problem", che richiede di individuare il punto nella griglia che si trova più vicino a un punto centrale fisso nello spazio, chiamato origine. Si tratta di un problema facile da risolvere in una griglia bidimensionale, ma se il numero di dimensioni aumenta, anche un computer quantistico non è più in grado di risolvere il problema in modo efficiente. Nuove applicazioni (come la crittografia completamente omomorfica, l'offuscamento del codice e la crittografia basata sugli attributi) sono state rese possibili utilizzando proprio la crittografia basata su reticoli. La maggior parte degli algoritmi di generazione delle chiavi che sfruttano questo codice sono relativamente semplici, efficienti e altamente parallelizzabili. Inoltre, la sicurezza di alcuni sistemi di questo tipo è già stata verificata. D'altra parte, si è dimostrato difficile fornire stime precise della sicurezza degli schemi reticolari anche rispetto alle tecniche di crittoanalisi note.

Crittografia basata su codici

Nel 1978, Robert McEliece ha introdotto un sistema crittografico che, a tutt'oggi, risulta non violato. Da allora sono stati proposti altri sistemi basati su codici per la correzione di errori. Sebbene relativamente veloci, la maggior parte delle primitive basate su codice soffre di dimensioni di chiave molto grandi. Le varianti più recenti hanno introdotto più struttura nei codici nel tentativo di ridurre le dimensioni delle chiavi, tuttavia le novità aggiunte hanno anche portato all'introduzione di nuovi attacchi, attualmente oggetto di studio.

Crittografia polinomiale multivariata

Questi schemi si basano sulla difficoltà di risolvere sistemi di polinomi multivariati su campi finiti. Negli ultimi decenni sono stati proposti diversi crittosistemi multivariati, molti dei quali si sono però rivelati inaffidabili. Sebbene siano state avanzate alcune proposte, la crittografia multivariata ha storicamente avuto più successo come approccio alla firma digitale.

Firme basate su hash

Si tratta di un algoritmo matematico che mappa dei dati di lunghezza arbitraria (*messaggio*) in una stringa binaria di dimensione fissa chiamata *valore di hash*, ma spesso indicata anche con il termine inglese *message digest* (o semplicemente *digest*). Tale funzione hash è progettata per essere unidirezionale (*one-way*), ovvero una funzione difficile da invertire: l'unico modo per rigenerare i dati di input dall'output di una funzione di hash ideale è quello di tentare una ricerca a forza-bruta di possibili input per vedere se vi è corrispondenza (match). In alternativa, si potrebbe utilizzare una tabella di hash corrispondenti.

La sicurezza dell'algoritmo, anche contro gli attacchi quantistici, è ben compresa. Tuttavia molti dei più efficienti schemi di firma basati su hash presentano l'inconveniente che il firmatario deve tenere traccia del numero esatto di messaggi firmati in precedenza e qualsiasi errore in questo record comporterà insicurezza. Un altro svantaggio è che si può produrre solo un numero limitato di firme. Questo può essere aumentato, fino al punto di essere potenzialmente illimitato, ma questo accresce anche la dimensione della firma.

A partire da questi concetti generali sulla crittografia post-quantum di seguito viene presentata una breve panoramica sui 3 schemi DSA finalisti del terzo round del NIST. Questi schemi, infatti, sono potenziali sostituti dell'ECDSA, ad esempio utilizzato in AnFRA, come discusso in precedenza.

FALCON: FAST-FOURIER LATTICE-BASED

La logica di progettazione di Falcon deriva da una semplice osservazione: quando si passa da firme basate su RSA o logaritmi discreti a firme post-quantum, la complessità della trasmissione in termini, ad esempio, di lunghezza delle chiavi, sarà probabilmente un problema più rilevante rispetto alla velocità di esecuzione. Infatti, molti schemi post-quantum hanno una semplice descrizione algebrica che li rende veloci, ma tutti richiedono chiavi e firme con dimensioni più grandi degli schemi pre-quantici.

Ci aspettiamo che tali problemi di prestazioni ostacolino la transizione da schemi pre-quantici a schemi post-quantum. Quindi durante la progettazione si è cercato di minimizzare la seguente quantità:

|Pk| + |sig| = (dimensione in bit della chiave pubblica) + (dimensione in bit di una firma).

Questo porta a considerare le firme basate sui reticoli, che riescono a mantenere sia |pk| che |sig| piuttosto piccoli, specialmente per i reticoli strutturati. Quando si tratta di firme basate su reticoli, ci sono essenzialmente due modelli: Fiat-Shamir o hash-and-sign. Entrambi gli schemi raggiungono livelli comparabili di compattezza, ma l'hash-and-sign ha proprietà interessanti: il framework GPV [13], che descrive come ottenere schemi di firma basati su reticoli hash-and-sign, è sicuro nei modelli ad oracolo classico e quantistico [13, 14]. Inoltre, gode di capacità di recupero dei messaggi [15]. Quindi è stata presa in considerazione questa struttura.

Successivamente, si è scelta una classe di reticoli crittografici per rappresentare questo framework. Una scelta quasi ottimale rispetto al principio di progettazione (la compattezza) sono i tralicci NTRU: essi permettono di ottenere un'implementazione compatta [16] del framework GPV. Inoltre, la loro struttura accelera molte operazioni di due ordini di grandezza.

Per finire, è stato ideato un nuovo campionatore di trapdoor che è asintoticamente veloce come il miglior campionatore generico di trapdoor attualmente in uso [17] e fornisce lo stesso livello di sicurezza del più affidabile campionatore [18].

Per una trattazione approfondita dell'argomento si può far riferimento a [19].

CRYSTALS-Dilithium

Ora verrà presentato lo schema di firma digitale Dilithium, la cui sicurezza si basa sulla difficoltà di trovare vettori brevi nei reticoli. Lo schema è stato progettato tenendo presenti i seguenti criteri:

Semplicità di implementazione. Gli schemi di firma basati su reticoli più compatti [20, 21] richiedono la generazione di campioni casuali segreti con distribuzione gaussiana discreta**.**

Generare tali campioni in modo che siano sicuri contro gli attacchi side-channel è molto difficile e può facilmente portare a implementazioni insicure, come dimostrato in [22, 23, 24]. Sebbene sia

possibile che un'implementazione molto attenta possa prevenire tali attacchi, è irragionevole presumere che uno schema distribuito universalmente contenente molti punti critici sia sempre implementato in maniera corretta. Dilithium utilizza quindi solo il campionamento uniforme, come originariamente proposto per le firme in [25, 26]. Inoltre, tutte le altre operazioni (come la moltiplicazione polinomiale e l'arrotondamento) sono facilmente implementate in tempo costante.

Conservatività dei parametri. Poiché si punta alla sicurezza a lungo termine, è stata analizzata l'applicabilità degli attacchi su reticolo da un punto di vista molto favorevole dell'attaccante. In particolare, si sono presi in considerazione algoritmi quantistici le cui caratteristiche di dimensione sono dello stesso ordine di quelle del tempo. Tali algoritmi non sono attualmente implementabili, ma viene considerata la possibilità che possano aversi dei miglioramenti in futuro.

Minimizzazione delle dimensioni della chiave pubblica e della firma. Poiché molte applicazioni richiedono la trasmissione sia della chiave pubblica che della firma, questo schema e stato progettato per ridurre al minimo la dimensione di questi parametri. Sotto il vincolo di evitare l'utilizzo del campionamento gaussiano discreto, per quanto è possibile sapere, Dilithium ha la più piccola dimensione di firma e di chiave pubblica di qualsiasi altro schema basato su reticolo con gli stessi livelli di sicurezza.

Modularità dei livelli di sicurezza. Le due operazioni che costituiscono la quasi totalità delle procedure di firma e di verifica sono l'espansione di una Extendable Output Function (XOF) (si utilizzano SHAKE-128 e SHAKE-256) e la moltiplicazione dall'anello polinomiale $Z_q[X]/(X^n + 1)$. Implementazioni altamente efficienti di tale algoritmo dovranno quindi ottimizzare queste operazioni e assicurarsi che vengano eseguite in tempo costante. Per tutti i livelli di sicurezza, il presente schema utilizza lo stesso anello con $q = 2^{23} - 2^{13} + 1$ e $n = 256$. Variare la sicurezza significa semplicemente fare più/meno operazioni su questo anello e variare l'espansione dello XOF. In altre parole, una volta ottenuta un'implementazione ottimizzata per un certo livello di sicurezza, è facile ottenere la stessa implementazione per un livello di sicurezza superiore/inferiore.

Per una trattazione approfondita dell'argomento si può far riferimento a [27]

RAINBOW

Rainbow appartiene alla famiglia dei crittosistemi a chiave pubblica multivariata, una delle principali famiglie di crittosistemi post-quantum. Rainbow è stato progettato nel 2004 da Jintai Ding e Dieter Schmidt e si basa sullo schema della firma Oil-Vinegar inventato da Jacques Patarin.

La sicurezza teorica di Rainbow si basa sulla risoluzione di un insieme di sistemi quadratici multivariati casuali. La teoria matematica alla base è la teoria dei polinomi multivariati è la geometria algebrica.

Rainbow offre firme molto piccole, di poche centinaia di bit (circa 528 bit = 66 byte per il livello di sicurezza NIST I), che sono molto più corte di quelle di altri schemi di firma post-quantum. Inoltre, poiché Rainbow utilizza solo operazioni semplici su campi finiti di piccole dimensioni, la generazione e la verifica della firma sono estremamente efficienti.
Per una trattazione approfondita dell'argomento si può far riferimento a [28].

VII. CONCLUSIONI

Il presente lavoro di tesi ha avuto per oggetto uno studio preliminare del problema della sicurezza dei protocolli di comunicazione in applicazioni spaziali, tenendo conto dell'avvento, ormai imminente, dei computer quantistici. Poiché tali computer metteranno a rischio soprattutto la sicurezza degli algoritmi di crittografia a chiave pubblica, si è focalizzata l'attenzione su di essi. Il punto di partenza per l'analisi è stato fornito dai documenti CCSDS. Questi hanno messo in evidenza che la crittografia asimmetrica viene prevalentemente utilizzata in ambito spaziale ai fini dell'autenticazione. Ciò perché gli algoritmi asimmetrici sono computazionalmente onerosi e necessitano di hardware molto performanti, caratteristica questa che nei velivoli spaziali non è, di norma, presente. Si predilige quindi la crittografia simmetrica, in quanto è più leggera e veloce e consente di conseguire gli obbiettivi di riservatezza, integrità e autenticità. Di contro, il problema dello scambio delle chiavi non è, in queste applicazioni, particolarmente rilevante.
Oltre allo scenario deep-space, è stato considerato un contesto di applicazione di satelliti geo-stazionari (LEO) per il SIN, analizzando nel dettaglio l'algoritmo AnFRA, il quale utilizza, per le operazioni di autenticazione, l'algoritmo di firma digitale ECDSA, fino ad oggi ritenuto inviolabile. Con l'avvento dei computer quantistici tale sicurezza non è più garantita; pertanto, con l'intento di proporre una alternativa post-quantum, si è analizzato ECDSA per comprenderne meglio le caratteristiche. E' noto dalla letteratura che una soluzione post-quantum che preserva l'utilizzo delle curve ellittiche consiste nell'adozione di una particolare famiglia di curve, dette 'superingolari', basate su gruppi non abeliani Con esse è possibile progettare schemi crittografici potenzialmente resistenti all'attacco di computer quantistici, in quanto su questa famiglia di curve non è possibile applicare l'algoritmo di Shor.
Successivamente, sono stati introdotti schemi del tutto alternativi, quali quelli selezionati dal NIST per il terzo round del procedimento finalizzato alla definizione dei nuovi strandard post-quantici. L'approfondimento di questi schemi, negli scenari ora definiti sarà oggetto di lavori futuri, ma fin d'ora è possibile affermare che lo sviluppo di algoritmi crittografici robusti per applicazioni spaziali andrà di pari passo con l'analogo sviluppo per applicazioni convenzionali, e che vi sono le premesse

perché i problemi messi in evidenza possano essere brillantemente risolti, anche tenendo conto delle peculiarità dell'ambito spaziale.

VIII. RIFERIMENTI

[1] Lily Chen, Stephen Jordan, Yi-Kai Liu, Dustin Moody, Rene Peralta, Ray Perlner, Daniel Smith-Tone, NISTIR 8105 Report on Post-Quantum Cryptography.

[2] CCSDS 350.1-G-2 Security Threats against Space December Missions, Informational Report, 2015 Issue 2.

[3] CCSDS 350.0-G-3 The Application of Security to CCSDS Protocols, Informational Report, Issue 3.

[4] CCSDS 352.0-B-2 CCSDS Cryptographic Algorithms, Recommended Standard, Issue 2.

[5] CCSDS 350.9-G-1 CCSDS Cryptographic Algorithms, Informational Report, Issue 1.

[6] CCSDS 350.6-G-1 Space Missions Key Management Concept, Informational Report, Issue 1.

[7] CCSDS 355.1-B-1 Space Data Link Security Protocol - Extended Procedures, Recommended Standard, Issue 1.

[8] Y. Jiang, C. Lin, X. Shen, and M. Shi, "Mutual authentication and key exchange protocols for roaming services in wireless mobile networks," *IEEE Trans. Wireless Commun.*, vol. 5, no. 9, pp. 2569–2577, Sep. 2006.

[9] P. Gope and T. Hwang, "Lightweight and energy-efficient mutual authentication and key agreement scheme with user anonymity for secure communication in global mobility networks," *IEEE Syst. J.*, vol. 10, no. 4, pp. 1370–1379, Dec. 2016.

[10] Qingyou Yang, Kaiping Xue, AnFRA: Anonymous and Fast Roaming Authentication for Space Information Network.

[11] ANSI X9.62-1998 Public Key Cryptography for The Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA).

[12] J. Wohlwend, "Elliptic curve cryptography: Pre and post quantum," MIT, Tech. Rep., 2016.

[13] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new crypto-graphic constructions. In Richard E. Ladner and Cynthia Dwork, editors, 40th ACM STOC, pp. 197–206, Victoria, BC, Canada, May 17–20, 2008. ACM Press.

[14] Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random oracles in a quantum world. In Dong Hoon Lee and Xiaoyun Wang, editors, ASI-ACRYPT 2011, volume 7073 of LNCS, pp. 41–69, Seoul, South Korea, December 4–8, 2011. Springer, Heidelberg, Germany.

[15] Rafaël del Pino, Vadim Lyubashevsky, and David Pointcheval. The whole is less than the sum of its parts: Constructing more efficient lattice-based AKEs. In Vassilis Zikas and Roberto De Prisco, editors, SCN 16, volume 9841 of LNCS, pp. 273–291, Amalfi, Italy, August 31 – September 2, 2016. Springer, Heidelberg, Germany.

[16] Léo Ducas, Vadim Lyubashevsky, and Thomas Prest. Efficient identity-based encryption over NTRU lattices. In Palash Sarkar and Tetsu Iwata, editors, ASIACRYPT 2014, Part II, volume 8874 of LNCS, pp. 22–41, Kaoshiung, Taiwan, R.O.C., December 7–11, 2014. Springer, Heidelberg, Germany.

[17] Chris Peikert. An efficient and parallel Gaussian sampler for lattices. In Tal Rabin, editor, CRYPTO 2010, volume 6223 of LNCS, pp. 80–97, Santa Barbara, CA, USA, August 15–19, 2010. Springer, Heidelberg, Germany.

[18] Chris Peikert. An efficient and parallel Gaussian sampler for lattices. In Tal Rabin, editor, CRYPTO 2010, volume 6223 of LNCS, pp. 80–97, Santa Barbara, CA, USA, August 15–19, 2010. Springer, Heidelberg, Germany.

[19] Pierre-Alain Fouque et al. 2017. FALCON: Fast-Fourier Lattice-based Compact Signatures over NTRU. Technical Report. National Institute of Standards and Technology.

[20] Léo Ducas, Alain Durmus, Tancrède Lepoint, and Vadim Lyubashevsky. Lattice signatures and bimodal gaussians. In CRYPTO (1), pp. 40–56, 2013.

[21] Léo Ducas, Vadim Lyubashevsky, and Thomas Prest. Efficient identity-based encryption over NTRU lattices. In ASIACRYPT, pp. 22–41, 2014.

[22] Leon Groot Bruinderink, Andreas Hülsing, Tanja Lange, and Yuval Yarom. Flush, gauss, and reload - A cache attack on the BLISS lattice-based signature scheme. In CHES, pp. 323–345, 2016.

[23] Thomas Espitau, Pierre-Alain Fouque, Benoît Gérard, and Mehdi Tibouchi. Side-channel attacks on BLISS lattice-based signatures: Exploiting branch tracing against strongswan and electromagnetic emanations in microcon- trollers. In CCS, pp. 1857–1874, 2017.

[24] Peter Pessl, Leon Groot Bruinderink, and Yuval Yarom. To BLISS-B or not to be: Attacking strongswan's implementation of post-quantum signatures. In CCS, pp. 1843–1855, 2017.

[25] Vadim Lyubashevsky. Fiat-Shamir with aborts: Applications to lattice and factoring-based signatures. In ASIACRYPT, pp. 598–616, 2009.

[26] Tim Güneysu, Vadim Lyubashevsky, and Thomas Pöppelmann. Practical lattice-based cryptography: A signature scheme for embedded systems. In CHES, pp. 530–547, 2012.

[27] L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, and D. Stehlè. CRYSTALS-Dilithium Algorithm Specifications and Supporting Documentation. Accessed: Mar. 30, 2019. [Online]. Available: https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-.

[28] Ding, J., Schmidt, D.: Rainbow, a new multivariable polynomial signature scheme. In: Ioannidis, J., Keromytis, A., Yung, M. (eds.) Applied Cryptography and Network Security: Third International Conference, ACNS 2005, New York, NY, USA, June 7-10, 2005. Proceedings. pp. 164–175. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.

IX. RINGRAZIAMENTI

Mi è doveroso dedicare questo spazio del mio elaborato alle persone che hanno contribuito, con il loro instancabile supporto, alla realizzazione dello stesso.

In primis, un ringraziamento speciale al mio relatore Prof. Chiaraluce per la sua immensa pazienza, per i suoi indispensabili consigli, per le conoscenze trasmesse durante tutto il percorso di stesura dell'elaborato.

Ringrazio infinitamente la mia famiglia che mi ha sempre sostenuto, appoggiando ogni mia decisione, fin dalla scelta del mio percorso di studi.

Grazie a tutti i miei amici per essere stati sempre presenti anche durante questa ultima fase del mio percorso di studi. Grazie per aver ascoltato i miei sfoghi, grazie per tutti i momenti di spensieratezza.

Un grazie di cuore alla mia amica e compagna di studi Michela con cui ho condiviso l'intero percorso universitario. È grazie a lei che ho superato i momenti più difficili. Senza i suoi consigli, non ce l'avrei mai fatta.

Infine, dedico questa tesi a me stesso, ai miei sacrifici e alla mia tenacia che mi hanno permesso di arrivare fin qui.