**Università degli Studi di Padova**

Dipartimento di Matematica "Tullio Levi-Civita"

Corso di Laurea in Informatica



# Formalizzazione di una Architettura per Robot Sociali in un Contesto di Assistenza Sanitaria Domiciliare

*Tesi di Laurea*


*Relatore*
Prof. Conti Mauro

*Correlatore*
Dott. Donadel Denis

*Laureando*
Soldà Matteo
*Matricola 1226319*



Anno Accademico 2023-2024

*Dedicato al Matteo bambino e alla sua passione per i computer.*

# Sommario

Il presente documento descrive il lavoro svolto durante il periodo di stage interno, della durata di circa trecentoventi ore, dal laureando Soldà Matteo presso l'Università degli Studi di Padova. Il tirocinio è stato svolto con la guida del Prof. Conti Mauro nelle vesti del proponente e con la collaborazione del Dott. Donadel Denis, facente parte del gruppo di ricerca SPRITZ del Dipartimento di Matematica dell'Università di Padova. Tutor interno al Consiglio del Corso di Studio è stato il Prof. Bresolin Davide.

Questa tesi si occupa di formalizzare una architettura per robot sociali, in affiancamento a dispositivi di Mobile Health, in un contesto di assistenza sanitaria domiciliare. Abbiamo quindi definito dei requisiti che una architettura per robot sociali dovrebbe rispettare per garantire sicurezza e privacy per tutte le entità coinvolte nella rete. In seguito abbiamo analizzato l'architettura formalizzata per verificare che ogni requisito fosse stato pienamente soddisfatto.

Il progetto è stato suddiviso in due parti: la prima dedicata allo studio delle architetture attualmente presenti sul mercato seguita dalla formalizzazione di una generica architettura per social robot in un contesto sanitario domiciliare utilizzando tecnologie innovative, quali la Cipher Text Policy Attribute Based Encryption e la 1-way pseudonymity, permettendo comunque modularità, rendendo quindi intercambiabile il layer di persistenza o la responsabilità nella gestione dei dati. La seconda parte, invece, è stata dedicata all'adattamento e all'implementazione dell'architettura a un caso di studio offerto da un'azienda.

*"Sic Parvis Magna"*

— Sir. Francis Drake

# Ringraziamenti

*Innanzitutto, vorrei esprimere la mia gratitudine al relatore Prof. Conti Mauro per l'aiuto e il sostegno fornitomi durante il tirocinio e la stesura della tesi.*

*Ringrazio Donadel Denis, correlatore di questa tesi, per l'immensa disponibilità e gentilezza con la quale si è messo a disposizione per aiutarmi nello sviluppo di questo ambizioso e innovativo progetto.*

*Ringrazio Giuliana e Fabio, i miei genitori, per avermi dato la possibilità di proseguire con i miei studi, essendomi sempre vicini, nei momenti felici e in quelli più bui. Ringrazio inoltre tutta la famiglia, sempre presente e interessata al mio percorso universitario.*

*Ringrazio Alice, la mia fidanzata, per essermi sempre vicina e presente anche durante gli studi, per avermi supportato e sopportato in ogni momento, con una delicatezza e un amore straordinari.*

*Ringrazio gli amici della Croce Rossa, soprattutto i colleghi del 1° Venerdì Notte e della 4° Domenica Mattina, presenti fin da prima dell'inizio del percorso universitario, per avermi aiutato a crescere e con i quali ho vissuto esperienze uniche e indimenticabili.*

*Ho infine il desiderio di ringraziare gli amici presenti già da prima dell'inizio della carriera universitaria. Ringrazio anche i colleghi accademici che per lungo o breve tempo hanno intrecciato il loro percorso universitario con il mio, rendendolo più stimolante, leggero e indimenticabile.*

*Padova, Febbraio 2024* Soldà Matteo

# Indice

# Elenco delle figure

# Elenco delle tabelle

# Capitolo 1

# Introduzione

*All'interno del capitolo verranno presentati in ordine: una introduzione al progetto, il gruppo SPRITZ, l'azienda proponente, l'organizzazione del testo e le convenzioni tipografiche.*

## 1.1 Introduzione al Progetto

La tecnologia può essere uno strumento potente per aiutare coloro che ne hanno bisogno. In tale scenario, i robot sociali possono essere impiegati per monitorare gli anziani a casa e i pazienti nelle strutture di cura, riducendo e filtrando l'onere sugli operatori socio-sanitari, sui medici e sugli infermieri. A causa dei requisiti di controllo remoto e dell'esposizione a una vasta gamma di dati altamente sensibili (quali ad esempio, informazioni mediche, posizioni degli utenti, abitudini degli utenti), la privacy e la sicurezza sono necessità fondamentali in tale ambiente.

Abbiamo iniziato il progetto analizzando e valutando la sicurezza delle architetture presenti nella letteratura per poi tentare di implementare la prima architettura sicura, generale, adattabile e rispettosa della privacy per gli ambienti abitati anche dai robot sociali in un caso di studio offerto da una azienda. Investigando attentamente gli attacchi, dimostriamo come l'architettura sia in grado di resistere, per design, alla maggior parte delle minacce. Infine, dimostriamo la fattibilità dell'implementazione di tale architettura con una Proof of Concept (PoC)[g] nel mondo reale.

## 1.2 Lo SPRITZ Group

Lo stage in questa tesi discusso è stato svolto in collaborazione con il gruppo SPRITZ dell'Università degli Studi di Padova. Questo gruppo, il cui nome è acronimo di Security and PRIvacy Through Zeal, è nato nel 2011 ed è guidato dal Prof. Conti Mauro con lo scopo di accogliere ricercatori che vogliano contribuire allo sviluppo di nuove tecnologie per la sicurezza e la privacy.


**Figura 1.1:** Logo del gruppo SPRITZ [1]

## 1.3 L'Azienda

Omitech Robot, facente parte del gruppo Omitech, è una realtà nata nel 2018 che ha iniziato un percorso di sperimentazione e ricerca nel campo della robotica, considerato che la robotica sociale è un settore in forte crescita e che può rappresentare uno strumento efficace per il business.
Tra le loro creazioni, nasce *Vivaldi*, una intelligenza artificiale che controlla e adatta il comportamento dei robot con gli umani e l'ambiente. Lo stage, nelle fasi finali è stato svolto in collaborazione con Omitech Robot, i quali hanno fornito un caso di studio relativo all'implementazione dell'architettura proposta per il monitoraggio dei pazienti anziani situati all'interno di una casa di cura.

**Figura 1.2:** Logo dell'Azienda Omitech Robot [2]

## 1.4 Organizzazione del testo

**Il secondo capitolo** descrive il background, l'idea, gli obiettivi e la pianificazione del progetto;

**Il terzo capitolo** analizza le soluzioni al momento esistenti nei rispettivi punti di forza e criticità;

**Il quarto capitolo** approfondisce il design della architettura e motiva le scelte progettuali effettuate;

**Il quinto capitolo** descrive l'adattamento di una architettura preesistente e spiega le scelte implementative effettuate;

**Il sesto capitolo** approfondisce come sono state mitigate o eliminate le vulnerabilità descritte dai *security requirements*;

**Nel settimo capitolo** vengono riassunti i risultati ottenuti.

Riguardo la stesura del testo, relativamente al documento sono state adottate le seguenti convenzioni tipografiche:

- gli acronimi, le abbreviazioni e i termini ambigui o di uso non comune menzionati vengono definiti nel glossario, situato alla fine del presente documento;
- per la prima occorrenza dei termini riportati nel glossario viene utilizzata la seguente nomenclatura: *parola*[g];
- i termini in lingua straniera o facenti parti del gergo tecnico sono evidenziati con il carattere *corsivo*.

# Capitolo 2

# Descrizione del Progetto

*In questo capitolo si tratterà il background degli argomenti fulcro di questa architettura, l'idea iniziale e gli obiettivi da raggiungere, oltre alla pianificazione del lavoro e alla relativa variazione.*

## 2.1 Background

In questa sezione verrà presentata una descrizione degli argomenti che saranno trattati in dettaglio nei capitoli successivi.

### 2.1.1 Robot Sociali

Nel vasto regno della letteratura contemporanea sui robot sociali [3]–[9], è evidente che l'attenzione predominante si concentra sui robot umanoidi, progettati per interagire in una maniera quanto più naturale con gli umani: comportamento noto anche come Human-Robot Interaction (HRI). Questi robot, che costituiscono una categoria avanzata di automi, rappresentano una manifestazione significativa del campo della robotica sociale, un dominio di ricerca interdisciplinare che amalgama principi fondamentali dalla robotica, dall'intelligenza artificiale e dalle scienze sociali [4], [7]–[16].
L'evoluzione dei robot sociali affonda le sue radici negli ultimi decenni, rispondendo alla crescente necessità di sistemi robotici in grado di comprendere e impegnarsi in modo significativo nella complessità delle interazioni sociali umane. I domini applicativi includono assistenza nelle attività quotidiane, supporto emotivo in contesti terapeutici, interazione sociale in ambienti pubblici e lo sviluppo di soluzioni di intrattenimento.
Gli obiettivi ampi e diversificati dei robot sociali sono radicati nel facilitare un'interazione più simile a quella umana tra la macchina e l'utente, con l'intento di migliorare la qualità della vita umana, promuovere i processi di apprendimento e fornire supporto emotivo quando necessario. Tuttavia, la sfida continua è affrontare questioni cruciali come la comprensione del contesto sociale, l'integrazione etica nelle interazioni uomo-robot e l'ottimizzazione delle capacità cognitive e comunicative dei robot per garantire un coinvolgimento più significativo e fruttuoso. Lo sviluppo costante del campo, attraverso la ricerca e l'innovazione, mira a superare queste sfide, posizionando i robot sociali come protagonisti chiave nella definizione del futuro delle interazioni uomo-sociali con l'intelligenza artificiale.

Le architetture formalizzate nel corpus esistente della letteratura sui robot umanoidi tendono a dividere il sistema robotico in due componenti fondamentali: l'agente e il server. Il primo, che costituisce il livello automatico, è tipicamente locale e gestisce compiti come la ricezione di stimoli esterni tramite sensori, attuatori e l'esecuzione di funzioni come Speech-to-Text (S2T), Text-to-Speech (T2S), movimento e Human-Computer Interaction (HCI). Allo stesso tempo, il server, che rappresenta il livello deliberativo, è tipicamente remoto e dotato di un albero decisionale, che consente al robot di prendere decisioni informate sulla base dei dati accumulati durante la sua fase operativa.
Contemporaneamente, queste configurazioni architettoniche sono spesso stratificate in *layer*, con un conteggio stimato medio che va da 3 a 5. In una tipica architettura composta da 4 strati, i livelli includono uno strato comportamentale, uno strato dedicato alle capacità, uno strato che rappresenta la coscienza, e infine, uno strato focalizzato sui comportamenti autonomi fondamentali. Questa struttura a strati mette in mostra la raffinatezza di questi sistemi robotici e sottolinea la loro capacità di interagire e rispondere al loro ambiente in modo sottile e adattabile.
Da un punto di vista tecnologico, i robot sociali integrano un insieme sofisticato di componenti, tra cui sensori avanzati (telecamere, microfoni e sensori di movimento), attuatori complessi (motori e altoparlanti), unità di elaborazione dati (processori e memoria) e software sofisticato basato sull'intelligenza artificiale. L'orchestrazione sinergica di questi elementi conferisce ai robot la capacità di percepire l'ambiente circostante, interpretare segnali visivi e uditivi umani, imparare dalle interazioni passate e rispondere con coerenza e pertinenza al contesto.
A un livello di astrazione più basso, oggigiorno esistono diverse definizioni di robot sociali, ma come riportato da uno studio di Oruma [17], le principali proprietà includono autonomia, rappresentazione fisica, capacità di percezione e risposta a stimoli ambientali, capacità di interazione con gli esseri umani, animali e con altri robot, e la capacità di comprendere e seguire regole sociali non scritte. Oltre a queste proprietà, i robot sociali devono anche possedere caratteristiche come percezione, cognizione, efficienza, interazione ed etica.
A livello fisico, ci sono sensori che possono essere interni o esterni. I sensori interni aiutano a mantenere le dinamiche interne e la stabilità del robot, mentre i sensori esterni sono responsabili della percezione dell'ambiente esterno, come visione, rilevamento e navigazione. In questa ricerca definiamo un robot sociale come un automa robotico

**Figura 2.1:** Sanbot Elf [18]

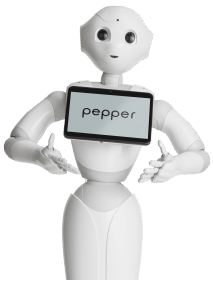


**Figura 2.2:** SoftBank Pepper [19]

collocato all'interno di una struttura sanitaria o di una casa. Per ridurre i costi e il peso

del robot, supponiamo che l'algoritmo decisionale (ad esempio, l'albero decisionale, modello di Machine Learning (ML)) che analizza i dati raccolti e gli output successivi alle azioni del robot sia ospitato in un server di controllo. Pertanto, il robot deve essere connesso a Internet per funzionare correttamente. Senza accesso alla rete, il robot può solo seguire procedure standard che generalmente sono inutili per il paziente. L'automa ha la capacità di muoversi grazie alle ruote, oltre alla presenza di eventuali arti robotici.
Considerando che per raggiungere il suo scopo deve consentire anche di effettuare videochiamate, avrà tra i suoi componenti una fotocamera, un altoparlante e uno schermo. Tra le sue funzionalità principali ricordiamo T2S e S2T che possono essere introdotte attraverso l'utilizzo di servizi forniti di terze parti.

### 2.1.2 Internet of Things

L'Internet of Things (IoT) rappresenta una rivoluzione tecnologica che ha profondamente trasformato il nostro modo di interagire con il mondo fisico.
Questa rete interconnessa di dispositivi intelligenti ha le sue origini negli anni '70, ma è solo negli ultimi due decenni che la tecnologia necessaria a rendere l'IoT una realtà è diventata ampiamente disponibile.
L'avvento di una connessione internet sempre più veloce e accessibile, insieme all'evoluzione di sensori più piccoli e efficienti, ha aperto la strada a una vasta gamma di applicazioni.
Le motivazioni alla base dell'adozione rapida dell'IoT sono molteplici. La crescente necessità di raccogliere dati in tempo reale per migliorare l'efficienza operativa e ottimizzare le risorse ha spinto molte industrie a implementare soluzioni IoT. Settori come quello sanitario, agricolo, logistico e manifatturiero hanno beneficiato notevolmente dall'integrazione di dispositivi intelligenti.
Inoltre, questi dispositivi offrono la possibilità di creare un ambiente più intelligente e connesso per migliorare la qualità della vita quotidiana. Dalle Smart Home[g] ai veicoli interconnessi, la visione di un mondo in cui ogni oggetto può comunicare e cooperare è diventata sempre più tangibile.

I dispositivi IoT sono composti da tre componenti chiave: sensori, attuatori e connettività.
I sensori rilevano dati dall'ambiente circostante, come temperatura, umidità, movimento o altri parametri, trasformandoli in segnali elettrici interpretabili. Gli attuatori, invece, eseguono azioni in risposta ai comandi ricevuti. La connettività, spesso fornita tramite tecnologie wireless, consente la trasmissione bidirezionale dei dati tra il dispositivo e una piattaforma di gestione. La piattaforma di gestione, solitamente basata sul Cloud Computing[g], svolge un ruolo cruciale nel processare e analizzare i dati raccolti dai dispositivi IoT.

Tra i dispositivi più famosi e conosciuti possiamo trovare gli assistenti digitali come Amazon Echo (2.3) e Google Nest (2.4), ma esistono alcuni strumenti ancora poco conosciuti, ad esempio dispositivi per la cura della persona come lo spazzolino da denti intelligente, dispositivi domestici come il frigorifero intelligente Samsung Family Hub (2.6) e dispositivi per la sicurezza come la serratura intelligente Samsung Smart Lock (2.5).

**Figura 2.3:** Amazon Echo Dot 4 [20]


**Figura 2.4:** Google Nest Hub [21]


**Figura 2.5:** Samsung Smart Lock [22]

**Figura 2.6:** Samsung Family Hub (Frigorifero) [23]

### 2.1.3 Mobile Health

I dispositivi di Mobile Health (mHealth)[g] sono dispositivi IoT elettronici portatili che possono essere indossati o impiantati nel corpo umano. Questi dispositivi sono in grado di raccogliere dati sullo stato di salute di un individuo e trasmetterli a un dispositivo mobile tramite una connessione wireless.

Questi dispositivi hanno iniziato a svilupparsi intorno agli anni 90, quando i primi cellulari hanno iniziato a diffondersi. I primi dispositivi di mHealth erano semplici e venivano utilizzati principalmente per fornire informazioni sulle condizioni di salute di un individuo, ad esempio i dispositivi per la registrazione della glicemia di una persona diabetica, come il FreeStyle Libre (2.10). Tuttavia, con l'evoluzione delle tecnologie *mobile*, i dispositivi di mHealth sono diventati più sofisticati e sono stati in grado di fornire informazioni più dettagliate e aggiornate (anche in real time) sullo stato di salute di un individuo, come per esempio gli *smartwatch* di Apple (2.7), Samsung (2.8) e Xiaomi (2.9).

**Figura 2.7:** Apple Watch Series 8 [24]

**Figura 2.8:** Samsung Galaxy Watch 4 [25]

**Figura 2.9:** Xiaomi SmartBand 6 [26]


**Figura 2.10:** FreeStyle Libre [27]

### 2.1.4 Protezione dei Dati

La protezione dei dati personali è diventata una preoccupazione cruciale nell'era digitale, con il crescente scambio di informazioni attraverso le reti elettroniche. Se esaminiamo la storia, possiamo tracciare le radici di questa preoccupazione sin dall'avvento delle prime forme di scrittura e comunicazione.
Con l'invenzione delle macchine cifranti nel $XV$ secolo da parte di Leon Battista Alberti, abbiamo assistito alle prime forme di crittografia, un elemento chiave nella protezione dei dati.

La crittografia implica la trasformazione delle informazioni in un formato sicuro, rendendole illeggibili agli utenti non autorizzati e garantendo che solo coloro che dispongono delle credenziali appropriate possano accedere o decifrare i dati: questa tecnologia ha avuto un ruolo fondamentale nella storia della sicurezza dei dati. Un punto di riferimento storico cruciale è l'Enigma , la macchina cifrante utilizzata dalle forze tedesche durante la Seconda Guerra Mondiale. Il lavoro degli Alleati nel rompere il codice Enigma, guidato da Alan Turing, ha dimostrato l'importanza della crittografia nella protezione delle comunicazioni sensibili.

Oggi, l'avanzamento tecnologico ha portato all'uso diffuso di algoritmi crittografici complessi per proteggere dati sensibili su Internet. Standard come Secure Sockets Layer (SSL) e Transport Layer Security (TLS) garantiscono la sicurezza delle transazioni online, assicurando che le informazioni personali degli utenti siano crittografate durante la trasmissione.

Con l'ascesa negli ultimissimi anni dei Big Data[g], la privacy differenziale è emersa come una risposta innovativa per proteggere l'identità degli individui. Questo approccio si basa sulla manipolazione dei dati in modo che le informazioni personali rimangano anonime anche quando vengono analizzate a livello aggregato. Un esempio storico

di questo concetto è il lavoro di George Polya, un matematico che ha introdotto il concetto di "conteggio differenziale" nel 1940.
L'implementazione di tecniche di privacy differenziale in contesti moderni, come l'analisi dei dati sanitari, sottolinea l'importanza di bilanciare la necessità di informazioni con la salvaguardia della privacy individuale.

Confrontando questi due metodi sui dati sanitari, si può notare che, mentre la crittografia si concentra sulla protezione dei dati tramite la modifica dell'intero messaggio in modo che non si possa leggere senza la chiave, la privacy differenziale si concentra sulla protezione dei dati tramite la modifica di alcuni elementi del messaggio in modo che non si possano fare collegamenti indiretti tra i dati presi in analisi e i dati precisi relativi al paziente.

La privacy differenziale, nel dettaglio, è una metodologia matematica che garantisce la privacy attraverso meccanismi che aggiungono rumore probabilistico[g] ai dati reali, come dimostrato dall'immagine 2.11. Tale garanzia è stata formalizzata per la prima

**Figura 2.11:** Esempio di Immagine con Rumore Probabilistico a Varie Intensità [28]

volta da Cynthia Dwork nel 2006 [29] ed è poi evoluta in forme più specifiche, come la epsilon-differential privacy [30]–[34], o con l'introduzione dell'Intelligenza Artificiale nella generazione di rumore solo nei campi dati in cui è necessario.

Parlando invece della crittografia, è possibile distinguere due tipi di algoritmi: a chiave simmetrica e a chiave asimmetrica, il cui funzionamento è essenzialmente uguale, come dimostrato dalla Figura 2.12, in cui la sola differenza risiede nel contenuto delle chiavi.
Nel primo caso, sia per crittare che per decrittare un blocco di dati, il mittente ed il destinatario utilizzano la stessa chiave. Tra gli algoritmi di spicco di questa categoria troviamo Data Encryption Standard (DES), Triple Data Encryption Standard (3DES), Advanced Encryption Standard (AES) e Rivest Cipher 4 (RC4).
Per quanto riguarda invece la crittografia a chiave asimmetrica si utilizzando due chiavi diverse, definite chiave privata (o segreta) e chiave pubblica: la prima serve per decrittare i dati e deve quindi essere mantenuta al sicuro, mentre la seconda (che è calcolata a partire dalla chiave privata) serve per crittare i dati e può quindi essere condivisa con chiunque. Tra gli algoritmi di spicco in questa categoria troviamo Rivest–Shamir–Adleman (RSA), Elliptic Curve Cryptography (ECC), Diffie-Hellman ed ElGamal.
Sorvolando sui motivi che rendono la crittografia a chiave asimmetrica più sicura

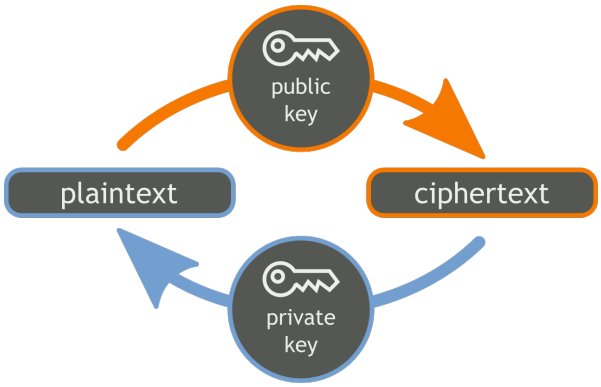


**Figura 2.12:** Funzionamento della Crittografia a Chiave Asimmetrica [35]

rispetto alla crittografia a chiave simmetrica (che non risulta di competenza di questa tesi e che viene ben documentato nell'attuale letteratura), bisogna sottolineare che al giorno d'oggi i due algoritmi principalmente usati son RSA e ECC.

RSA è un algoritmo di crittografia asimmetrica inventato nel 1977, con il suo acronimo che indica i nomi dei suoi tre creatori: Ronald Rivest, Adi Shamir e Leonard Adleman.
Questo algoritmo si basa sulla sua elevata complessità computazionale dovuta alla fattorizzazione dei numeri primi, in cui la chiave pubblica deriva dalla chiave privata.
D'altra parte, ECC è un algoritmo di crittografia a chiave pubblica inventato nel 1985 da Neal Koblitz e Victor S. Miller ed è basato sulla fattorizzazione di numeri interi attraverso curve ellittiche definite su campi finiti.
Come riportato in alcuni documenti, la crittografia RSA è ampiamente adottata e consolidata. I suoi vantaggi includono la maturità, l'ampia implementazione e la resistenza agli attacchi quantistici. Tuttavia, lo svantaggio principale risiede nella lunghezza delle chiavi, che può essere significativa per garantire una sicurezza adeguata.
In contrasto, la crittografia ECC offre livelli di sicurezza paragonabili a quelli di RSA con chiavi notevolmente più corte. Ciò comporta risparmi in termini di risorse computazionali e larghezza di banda.Tuttavia, la principale limitazione è la relativa novità della tecnologia, che potrebbe sollevare preoccupazioni sulla sua maturità e adozione diffusa.

Tra le nuove forme di crittografia appare la Attribute-Based Encryption (ABE)[g], che si basa sulla crittografia asimmetrica e permette di definire politiche di accesso ai dati, così da poter garantire l'anonimia dei dati, permettendo comunque un accesso granulare ai dati stessi, evitando di dover crittare i dati con una moltitudine di chiavi pubbliche.
Esistono due varianti principali di ABE: Key-Policy Attribute-Based Encryption (KP-ABE)[g] e Ciphertext-Policy Attribute-Based Encryption (CP-ABE)[g].
La differenza essenziale risede nel fatto che KP-ABE basa i privilegi di accesso sugli attributi della chiave, al contrario di CP-ABE che basa i privilegi di accesso sugli attributi dei dati, come mostrato in Figura 2.13. La crittografia basata su politiche di cifratura con attributi (CP-ABE) ha le sue radici nella crescente complessità delle esigenze di sicurezza nel panorama digitale.
Questo paradigma crittografico è emerso nel contesto della necessità di sviluppare

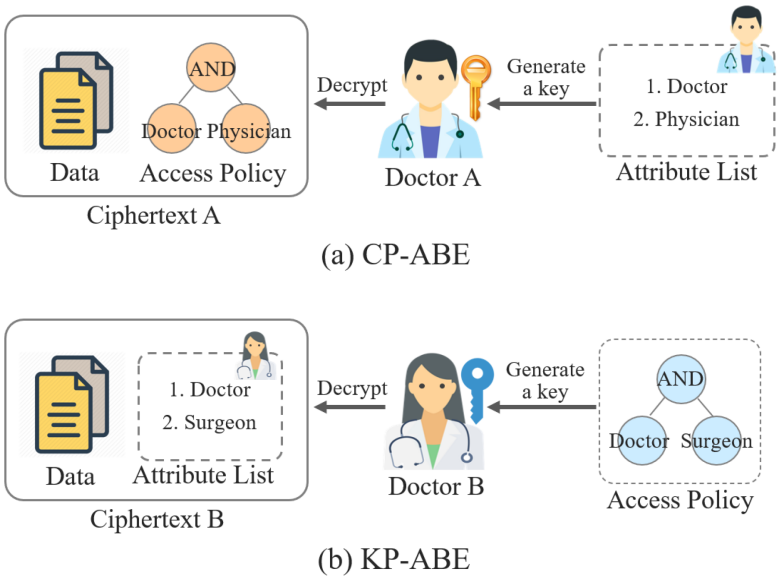


**Figura 2.13:** Differenze tra KP-ABE e CP-ABE [36]

sistemi più flessibili e adattabili per garantire la riservatezza dei dati, specialmente in scenari in cui la gestione delle chiavi tradizionali era limitante.
La storia della CP-ABE può essere fatta risalire ai primi anni del 21° secolo, quando gli studiosi della sicurezza informatica hanno iniziato ad esplorare nuove metodologie per affrontare sfide sempre più complesse.
Punto chiave dello sviluppo di questo meccanismo di crittografia furono Sahaj e Waters, che nel 2005 hanno pubblicato un articolo che ha introdotto il concetto di crittografia basata su attributi, di cui si può visualizzare un esempio nella Figura 2.14.
Da quel momento la crittografia CP-ABE ha subito evoluzioni e rifiniture significative, grazie anche al contributo di numerosi ricercatori nel campo della crittografia.

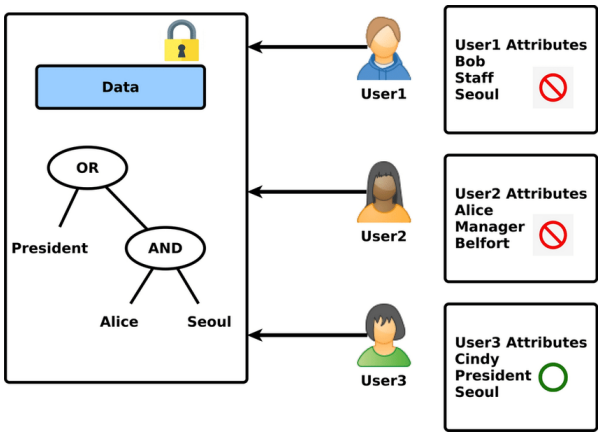


**Figura 2.14:** Caso Esempio di CP-ABE [37]

Sempre nel campo della crittografia, tipicamente queste tecniche sono affiancante e coadiuvate da dei metodi di hashing[g], che permettono di ottenere un codice univoco per un determinato input, che può essere utilizzato per verificare l'integrità dei dati.

## 2.2 L'Idea e gli Obiettivi

Gli obiettivi principali del tirocinio, in breve, sono stati quella della formalizzazione di una architettura sicura per robot sociali in un contento di *healtcare* e la sua implementazione in un caso di studio fornito da una azienda padovana del settore.

Si farà riferimento ai requisiti secondo le seguenti notazioni:

- *O* per i requisiti obbligatori, vincolanti in quanto obiettivo primario richiesto dal committente;
- *D* per i requisiti desiderabili, non vincolanti o strettamente necessari, ma dal riconoscibile valore aggiunto;
- *F* per i requisiti facoltativi, rappresentanti valore aggiunto non strettamente competitivo.

Le sigle precedentemente indicate all'interno della Tabella 2.1 saranno seguite da una coppia sequenziale di numeri, identificativo del requisito.

| ID | Descrizione |
|---|---|
| RO-01 | Studio del background della verifica formale dei flussi di dati |
| RO-02 | Studio del background dei dispositivi medici robotici domiciliari e dati gestiti |
| RO-03 | Studio del background dell'architettura dei robot sociali nella letteratura |
| RO-04 | Studio del background dell'architettura generale |
| RO-05 | Progettazione di una architettura sicura generica |
| RO-06 | Implementazione di una architettura sicura generica |
| RO-07 | Verifica dell'architettura sicura generica |
| RD-01 | Analisi dei flussi derivanti dalla architettura preesistente |
| RD-02 | Adattamento dell'architettura sicura generica all'architettura preesistente |
| RF-01 | Formalizzazione dei flussi di dati generati dalla architettura preesistente |

**Tabella 2.1:** Tabella degli Obiettivi

## 2.3 Pianificazione del Lavoro

Il periodo di svolgimento del progetto è stato dal 16 Ottobre 2023 all'8 Dicembre 2023, con l'aggiunta di 4 giorni a causa di alcune assenze, per un totale di 320 ore suddivise in 8 settimane lavorative full-time.

La pianificazione delle attività è riportata nella Tabella 2.2.

| Durata in ore | Descrizione dell'attività |
|---|---|
| **45** | **Background** |
| *15* | *Verifica formale dei flussi di dati* |
| *15* | *Studio dei dispositivi medici robotici domiciliari* |
| *15* | *Studio dell'architettura dei robot sociali nella letteratura* |
| **70** | **Studio dell'Architettura Preesistente** |
| *35* | *Studio dell'architettura generale* |
| *35* | *Identificazione dei flussi dati e definizione formale* |
| **110** | **Progettazione di un'Architettura Sicura Generica** |
| **45** | **Adattamento dell'Architettura Sicura all'Architettura Preesistente** |
| **50** | **Implementazione dell'Architettura Sicura** |
| **Totale ore** | **320** |

**Tabella 2.2:** Tabella della Pianificazione Oraria

## 2.4 Variazione degli Obiettivi Finali e della Pianificazione

A causa di alcune variazioni nella pianificazione del lavoro, si è reso necessario modificare alcuni obiettivi e la pianificazione oraria come di seguito riportato:

- Rimosso il requisito RO-01 in quanto non più necessario;
- Rimosso il requisito RO-04 in quanto non disponibile una architettura generale di riferimento, però per i quali sono state analizzate due architetture formalizzate per scopi diversi nel Capitolo 3;
- Rimosso il requisito RO-07 in quanto la verifica e la validazione della architettura non sono fattibili in quanto è stato proposto al massimo uno studio di fattibilità, come richiesto dalla azienda proponente, nonostante l'esistenza di una base teorica per l'architettura da loro progettata;
- Rimosso il requisito RD-02 in quanto non fattibile secondo quanto riferito nel punto sopra;
- Aggiunto il requisito RO-08: Studio del background delle blockchain e degli *smart contract*;
- Aggiunto il requisito RO-09: Studio del background della privacy e della crittografia;
- Aggiunto il requisito RF-02: Stesura del paper di ricerca;

Per questo motivo, nella Tabella 2.3 sono riportati gli obiettivi aggiornati. Data la modifica degli obiettivi, anche la pianificazione oraria ha subito dei cambiamenti, come riportato nella Tabella 2.4.

## 2.5 Strumenti Utilizzati

Di seguito verranno elencati i vari strumenti utilizzati durante lo svolgimento del progetto, ordinati alfabeticamente.

| ID | Descrizione |
|---|---|
| RO-02 | Studio del background dei dispositivi medici robotici domiciliari e dati gestiti |
| RO-03 | Studio del background dell'architettura dei robot sociali nella letteratura |
| RO-05 | Progettazione di una architettura sicura generica |
| RO-06 | Implementazione di una architettura sicura generica |
| RO-08 | Studio del background delle blockchain e degli *smart contract* |
| RO-09 | Studio del background della privacy e della crittografia |
| RD-01 | Analisi dei flussi derivanti dalla architettura proposta |
| RF-01 | Formalizzazione dei flussi di dati generati dalla architettura proposta |
| RF-02 | Stesura del paper di ricerca |

**Tabella 2.3:** Tabella Aggiornata degli Obiettivi

| Durata in ore | Descrizione dell'attività |
|---|---|
| **60** | **Background** |
| *15* | *Studio dei dispositivi medici robotici domiciliari* |
| *15* | *Studio dell'architettura dei robot sociali nella letteratura* |
| *15* | *Studio delle blockchain e degli smart contract* |
| *15* | *Studio della privacy e della crittografia* |
| **30** | **Studio della Architettura Proposta** |
| *25* | *Studio dell'architettura proposta* |
| *5* | *Identificazione dei flussi dati e definizione formale* |
| **130** | **Progettazione di un'Architettura Sicura Generica** |
| **60** | **Adattamento dell'Architettura Sicura all'Architettura Proposta** |
| **20** | **Studio di Fattibilità per l'Architettura Proposta** |
| **20** | **Stesura Paper** |
| **Totale ore** | **320** |

**Tabella 2.4:** Tabella Aggiornata della Pianificazione Oraria

## Draw.io

Draw.io è un'applicazione web gratuita per la creazione di diagrammi e disegni tecnici. Offre un'ampia varietà di forme e strumenti per la realizzazione di diagrammi flowchart, organigrammi, mappe concettuali e altri tipi di rappresentazioni visive. Draw.io è accessibile direttamente da un browser senza richiedere il download di software aggiuntivo. La sua interfaccia intuitiva e la possibilità di esportare i diagrammi in diversi formati lo rendono una scelta popolare per la creazione di grafici e schemi in ambito professionale e accademico.
Durante il tirocinio è stato utilizzato per la creazione dei diagrammi di flusso e per la rappresentazione grafica dell'architettura.

## Google Drive

Google Drive è un servizio di cloud storage offerto da Google, che consente agli utenti di archiviare e condividere file online. Consente di caricare documenti, foto, video e altri tipi di file su un server remoto, consentendo l'accesso da qualsiasi dispositivo con

**Figura 2.15:** Logo di Draw.io [38]

connessione a Internet. Google Drive offre anche strumenti di collaborazione in tempo reale, come Google Docs, Sheets e Slides, che consentono agli utenti di lavorare insieme su documenti, fogli di calcolo e presentazioni in modalità sincrona. Questo servizio è ampiamente utilizzato per la sua facilità d'uso, la sincronizzazione automatica e la possibilità di condividere facilmente i file con altri utenti.
Durante il tirocinio è stato utilizzato per condividere i file tra i membri del gruppo.

**Figura 2.16:** Logo di Google Drive [39]

### Google Docs

Google Docs è un'applicazione di elaborazione testi basata su cloud offerta da Google. Consente agli utenti di creare, modificare e condividere documenti online. Grazie alla sua natura basata sul cloud, i documenti sono accessibili da qualsiasi dispositivo con connessione a Internet. Google Docs offre strumenti di formattazione e modifica avanzati, oltre a funzionalità di collaborazione in tempo reale, consentendo a più persone di lavorare contemporaneamente sugli stessi documenti.
Durante il tirocinio è stato utilizzato per la stesura della documentazione e per il mantenimento di un diario giornaliero.

**Figura 2.17:** Logo di Google Docs [40]

### Google Scholar

Google Scholar è un motore di ricerca accademico fornito da Google. Si concentra sulla ricerca di articoli scientifici, tesi di dottorato, libri, conferenze e brevetti accademici. Gli utenti possono utilizzarlo per ottenere accesso a fonti accademiche e scientifiche in diversi campi di studio. Google Scholar facilita la ricerca di letteratura accademica, fornendo citazioni e link alle pubblicazioni originali.

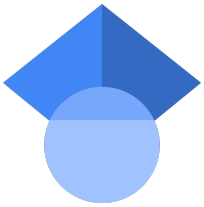

**Figura 2.18:** Logo di Google Scholar [41]

### Google Sheets

Google Sheets è un'applicazione di fogli di calcolo basata su cloud sviluppata da Google. Consente agli utenti di creare, modificare e condividere fogli di calcolo online. Esso offre una vasta gamma di funzionalità di elaborazione dati, formule e grafici. Grazie alla sua natura basata sul cloud, i fogli di calcolo possono essere accessibili da qualsiasi dispositivo con connessione a Internet. Inoltre, Google Sheets supporta la collaborazione in tempo reale, consentendo a più utenti di lavorare contemporaneamente sullo stesso foglio di calcolo.

**Figura 2.19:** Logo di Google Sheets [42]

### Google Slides

Google Slides è un'applicazione di presentazione basata su cloud sviluppata da Google. Consente agli utenti di creare, modificare e condividere presentazioni online. Esso offre strumenti per aggiungere diapositive, testo, immagini, grafici e altro ancora. Grazie alla sua natura basata sul cloud, le presentazioni possono essere accessibili da qualsiasi dispositivo con connessione a Internet. Inoltre, supporta la collaborazione in tempo reale, permettendo a più utenti di lavorare insieme sulla stessa presentazione.

**Figura 2.20:** Logo di Google Slides [43]

### Mendeley

Mendeley è un gestore di riferimenti bibliografici e una piattaforma di collaborazione accademica. Consente agli utenti di organizzare, annotare e condividere ricerche scientifiche. Mendeley facilita anche la creazione di bibliografie e citazioni, semplificando il processo di gestione delle fonti bibliografiche per gli studenti, i ricercatori e gli accademici. La piattaforma offre anche strumenti per scoprire e accedere a articoli accademici.

**Figura 2.21:** Logo di Mendeley [44]

### Oracle VM VirtualBox

Oracle VM VirtualBox è un software di virtualizzazione gratuito e open-source che consente agli utenti di creare e gestire macchine virtuali su un host fisico. Supporta una vasta gamma di sistemi operativi guest, consentendo agli utenti di eseguire più ambienti operativi su un singolo computer. VirtualBox offre funzionalità avanzate, come la snapshot delle macchine virtuali, che consente di salvare uno stato specifico e ripristinarlo in qualsiasi momento. È utilizzato sia per scopi di sviluppo e test che per l'esecuzione di applicazioni in ambienti isolati.
Durante il tirocinio è stato utilizzato per l'implementazione del PoC.


**Figura 2.22:** Logo di Oracle VM VirtualBox [45]

### Overleaf

Overleaf è una piattaforma di scrittura collaborativa basata su cloud per la creazione di documenti scientifici e accademici in LaTeX. Consente a più autori di lavorare

simultaneamente su un documento, fornendo strumenti per la gestione di progetti, revisione del testo e integrazione di formule matematiche complesse. Overleaf semplifica il processo di scrittura e formattazione, eliminando la necessità di installare e gestire un ambiente LaTeXlocalmente. È ampiamente utilizzato nella comunità accademica per la produzione collaborativa di articoli, tesi e altri documenti scientifici.
Durante il tirocinio è stato utilizzato per la stesura del paper di cui si parlerà nel Capitolo 7.


**Figura 2.23:** Logo di Overleaf [46]

### Visual Studio Code

Visual Studio Code è un editor di codice sorgente gratuito e open-source sviluppato da Microsoft. È leggero, altamente personalizzabile e supporta una vasta gamma di linguaggi di programmazione. Dotato di funzionalità avanzate come il completamento automatico, il debug integrato, il controllo di versione e un ricco ecosistema di estensioni, Visual Studio Code è ampiamente utilizzato dagli sviluppatori per la scrittura e la manutenzione del codice in modo efficiente. La sua interfaccia utente intuitiva e le potenti funzionalità lo rendono una scelta popolare nell'ambito dello sviluppo software.
Durante il tirocinio è stato utilizzato per l'implementazione del PoC.

**Figura 2.24:** Logo di Visual Studio Code [47]

# Capitolo 3

# Soluzioni Esistenti

*In questo capitolo verranno presentate le varie proposte presenti in letteratura per quanto riguarda le architetture di robot sociali, le architetture per la gestione di dati sanitari e per quanto riguarda gli schemi privacy-preserving utilizzati in ambito sanitario.*

Nella letteratura riguardante i robot sociali e il loro utilizzo nell'ambiente sanitario sono presenti molti studi sistematici che riguardano i robot sociali negli ospedali [6], molto spesso soffermandosi nella riduzione dello stress e del dolore [4], [10], [48], come supporto per i bambini autistici [8], [9] e negli ultimi anni a supporto degli anziani che vivono da soli o nelle strutture a seguito del COrona VIrus Disease 2019 (COVID-19)[49].

Gli algoritmi di controllo e HRI sono tra i componenti più complicati di un robot sociale [14], [50]. Molti articoli affrontano il problema in modi diversi, ad esempio, con architetture bio-ispirate [13].
Un'altra parte importante è l'architettura singola del robot sociale. È un argomento ampiamente discusso nella letteratura [50]–[52], sia con implementazioni software [5], [53]–[55] che requisiti hardware [56], [57].

Alcuni articoli discutono invece la connessione dei robot ad altri nodi e altri requisiti di rete per uno scenario del genere. Bonaccorsi et al. [48], [58] hanno proposto una soluzione robotica basata sul cloud e introdotto il paradigma "Robot-as-a-Service" nel campo. Loza et al. [59] hanno brevemente discusso un servizio web basato su Cloud Computing attraverso il quale gli utenti possono interagire con i robot sociali a distanza. Tuttavia, entrambi questi articoli riconoscono la sicurezza come un requisito senza approfondirne la discussione. Solo alcuni articoli indagano gli aspetti di sicurezza e privacy dell'ambiente del robot sociale, anche se è sentito come una necessità dagli utenti [60].
Miller et al. [61] hanno analizzato gli aspetti di sicurezza di un robot commerciale, concentrandosi sul robot fisico e non sull'architettura complessiva della rete. Il ricercatore presenta alcuni possibili attacchi che sfruttano porte aperte e protocolli di comunicazione utilizzati su un robot sociale in fase di test. Ricerche simili sono state condotte da Denning [62] che propone un'analisi di tre piccoli robot domestici. D'altro canto, Oruma et al. [17] propongono una ricerca empirica sul panorama delle minacce dei robot sociali che operano in luoghi pubblici. Nel nostro studio, consideriamo luoghi privati come case o strutture di cura, che presentano un modello di minaccia molto diverso. Mentre entrambi gli articoli indagano sulle vulnerabilità di un singolo robot

sociale, non discutono possibili problemi di sicurezza in una rete.
Al contrario, altri articoli [63]–[65] hanno discusso architetture per la salute generale, sia centralizzate che decentralizzate, fornendo un'analisi completa della sicurezza. Tuttavia, i robot sociali non sono stati menzionati esplicitamente negli articoli e le architetture proposte non sono automaticamente applicabili allo scenario del robot sociale.

Le implicazioni sulla privacy dei robot sociali sono state indagate anche a causa della sensibilità dei dati trasmessi. Infatti, questo campo sfida non solo la privacy delle informazioni degli utenti, ma influisce anche sulla loro privacy fisica, psicologica e sociale a causa dell'autonomia e del potenziale legame sociale dei robot sociali [66]. Tramite uno studio è stato dimostrato come alcune persone di un ufficio, ritenendo innocuo un robot sociale, gli avessero permesso di entrare in dei locali il cui accesso era limtato, fornendogli inoltre alcune informazioni sensibili [7]. Sono state indagate anche le esigenze di privacy su un singolo robot. La comprensione delle preferenze degli utenti in termini di avvisi sulla privacy è stata esaminata attraverso sperimentazioni empiriche [15]. Heuer et al. [67] hanno discusso dell'impiego del paradigma Privacy By Design nello sviluppo di un aspirapolvere automatico, fornendo un framework per aiutare gli sviluppatori a implementare strategie simili in altri contesti.

Oltre ai miglioramenti tecnologici in materia di sicurezza, dovrebbero essere considerati anche gli aspetti legali. Dal momento che i paesi stanno emanando normative per proteggere l'uso e la conservazione dei dati sensibili degli utenti [68], [69], i robot sociali dovrebbero conformarsi ad essi. Pagallo [70] ha discusso degli aspetti legali sulla privacy nel contesto dei robot per uso personale e domestico con l'uso dei servizi di Cloud Computing. Subramanian et al. [11] hanno fornito considerazioni su questioni di sicurezza, privacy e politiche nel campo dei robot sociali, specialmente quando è impiegata l'intelligenza artificiale.

# Capitolo 4

# SSRA: Secure Social Robot Architecture

*In questo capitolo si presenterà l'architettura proposta e se ne discuteranno nel dettaglio le componenti, i processi e i punti di forza.*

## 4.1 Threat Model

Nel modello che verrà presentato, presupposto fondamentale è che si possa stabilire una connessione sicura tra l'utente e il server, così da permettere di scambiare in modo sicuro le prime chiavi. Questo può essere fatto *offline* oppure offband[g] utilizzando altri canali di comunicazione sicuri già stabiliti tra le due parti.

In un'architettura così complessa, molti collegamenti e nodi diversi sono esposti agli attori minacciosi. In questo contesto, abbiamo preso in considerazione un attaccante che può ottenere accesso alla rete in due modi diversi, ovvero compromettendo un nodo o un collegamento. Per compromettere un nodo, un attaccante può sfruttare zero-day vulnerabilities[g] nell'hardware o nel software installato nel robot sociale [61] o lanciare attacchi di phishing mirati a un utente che sta utilizzando il laptop per monitorare un paziente [71]. Similmente a quest'ultimo caso, consideriamo anche un attaccante, che è un nodo legittimo ma curioso nella rete, come un utente malintenzionato che desidera ottenere l'identità di un paziente per commettere estorsione. Consideriamo che tutti i nodi possono essere compromessi, ma che la *master key* è salvata nei Trusted Platform Module (TPM)[g] e, quindi, non recuperabile da un attaccante con controllo sul server principale. D'altro canto, un attaccante potrebbe intercettare, effettuare un attacco Man In The Middle (MITM)[g] (il cui funzionamento è dimostrato in maniera semplice nella Figura 4.1) o disturbare una connessione tra due nodi nella rete. Ad esempio, un attaccante potrebbe sfruttare un DNS Cache Poisoning[g] [72] (esplicato dalla Figura 4.2) per compromettere una connessione che scorre su Internet o lanciare jamming attack[g] contro una rete wifi per bloccare una connessione tra un robot e il server [73]. Consideriamo un attaccante incapace di violare schemi di crittografia all'avanguardia, come gli algoritmi approvati dal National Institute of Standards and Technology (NIST)[g] impiegati in TLS [76]. Inoltre, non affrontiamo la sicurezza dei dispositivi mHealth e la connessione tra essi e il robot poiché è un argomento molto

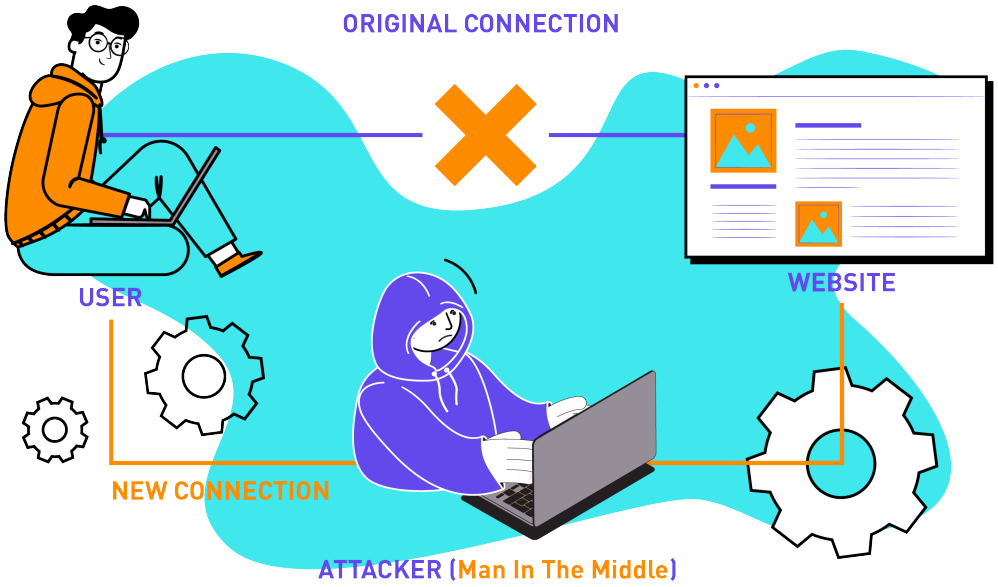


**Figura 4.1:** Esempio di Attacco Man in the Middle [74]

studiato e ritenuto non di interesse per il progetto [77]–[79].

## 4.2 Security Requirements

I robot sociali richiedono molte interconnessioni e comunicazioni tra diverse parti per svolgere compiti in modo completo e utile. Un tale scenario complesso presenta diverse minacce alla sicurezza delle comunicazioni e alla privacy dei dati, aspetti fondamentali da affrontare sin dall'inizio sviluppando un'architettura che contenga misure di sicurezza progettate *ad hoc*.
Inoltre, come con ogni altro progresso tecnologico, un passo importante da considerare è la fiducia e l'accettazione a livello sociale del nuovo servizio. Tra molti fattori [80], la sicurezza e la privacy dei dati svolgono un ruolo importante, specialmente quando si tratta di dati sensibili utente come quelli nel settore sanitario.
In aggiunta, i paesi impongono regolamentazioni per garantire queste proprietà ai dispositivi presenti sul mercato [68].

Per affrontare queste problematiche, formalizziamo sette (7) requisiti di sicurezza, che dovrebbero essere affrontati da un'architettura per i robot sociali, descritti di seguito:

1. **Integrità dei Dati.** Assicurazione dell'accuratezza e coerenza dei dati durante l'intero ciclo di vita. In particolare, è essenziale che i dati raccolti dai sensori non siano alterati durante la trasmissione per evitare azioni errate da parte del robot;

2. **Confidenzialità dei Dati.** Deve essere impedito l'accesso non autorizzato ai dati durante l'intero ciclo di vita dei dati;

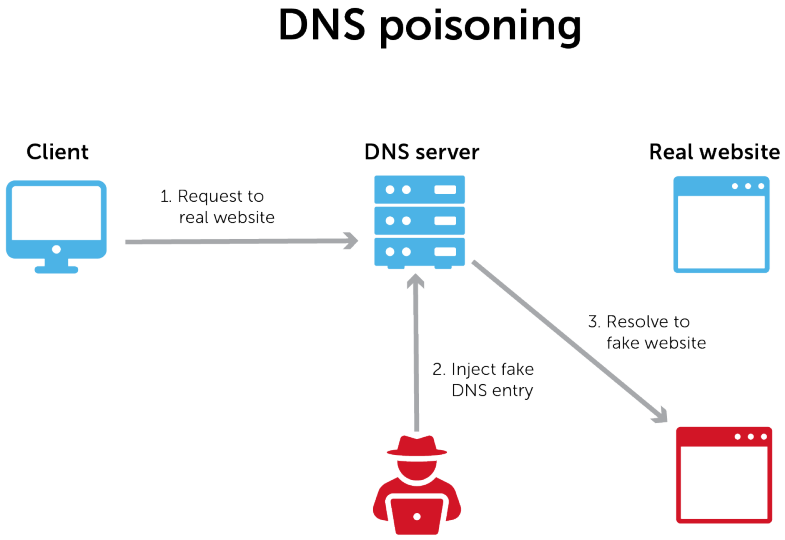


**Figura 4.2:** Esempio di Attacco DNS Cache Poisoning [75]

3. **Controllo dei Danneggiamenti.** Un attaccante che compromette un singolo nodo o collegamento non può intensificare e danneggiare porzioni più ampie della rete, tranne per le legittime capacità del dispositivo compromesso;

4. **Resistenza all'Impersonificazione.** Un attaccante non dovrebbe essere in grado di impersonare un altro nodo per comunicare con altri nodi legittimi;

5. **Continuità.** La rete dovrebbe essere sempre online o con tempi offline a bassa durata. In nessuna circostanza il robot dovrebbe agire in modo maligno o che possa mettere a repentaglio la sicurezza del paziente;

6. **Assenza di Tracciabilità.** Solo gli utenti autorizzati dovrebbero essere in grado di collegare i dati ai pazienti reali. Un attaccante che riesce a rubare dati da qualsiasi nodo o collegamento non dovrebbe essere in grado di risalire alle informazioni del paziente;

7. **Protezione dai Replay Attack.** Un vecchio blocco di dati non dovrebbe essere accettato una seconda volta in una rete.

## 4.3 Architettura

Partendo dai lavori presenti nella letteratura, abbiamo proposto Secure Social Robot Architecture (SSRA), che soddisfa tutti i requisiti discussi nella Sezione 4.2.

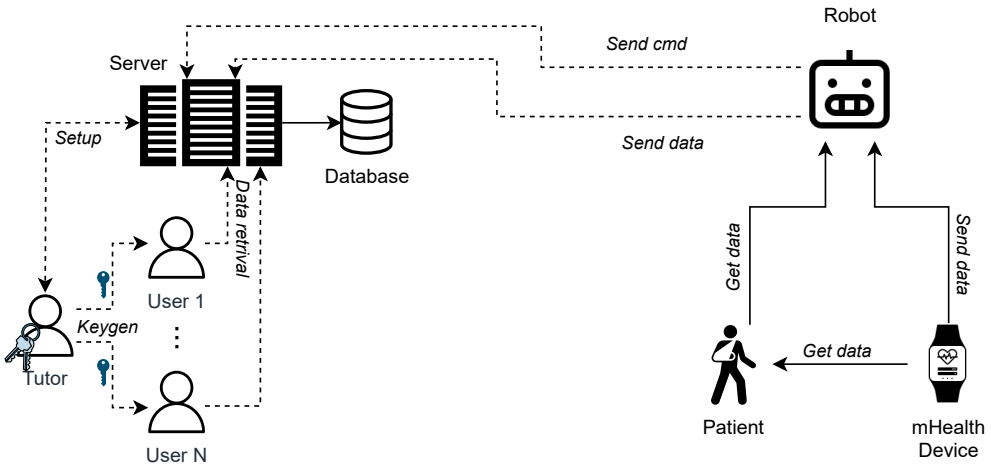


**Figura 4.3:** Diagramma di SSRA

L'architettura proposta è composta da diversi componenti. La persona assistita, solitamente una persona anziana con capacità motorie e di comunicazione conservate ma che richiede comunque una supervisione continua, è chiamata Paziente ($P_i$).
Ogni Paziente è monitorato da uno o più Robot Sociali ($R_i^1, \ldots, R_i^j$), automi dotati di capacità di movimento e sensori audio-visivi posizionati nella residenza del paziente, che possono essere collegati tramite una connessione sicura a dispositivi di mHealth. Si tratta di dispositivi indossabili utilizzati dal Paziente per rilevare un sottoinsieme dei parametri vitali tramite metodi non invasivi (ad esempio, frequenza cardiaca, temperatura, saturazione del sangue). Questi inviano un flusso di dati al Robot attraverso una connessione sicura, aspetto che non è oggetto di questa ricerca in cui consideriamo i dispositivi mobili come sensori esterni del Robot. Tuttavia, nella letteratura sono presenti numerosi articoli che affrontano le preoccupazioni di sicurezza e privacy di questi dispositivi [77]–[79].
Per semplificare la notazione, a volte ci riferiremo a $R_i^1$ come $R_i$.
I Robot inviano dati a un Server ($S$), che è connesso a un database ($DB$).
Ogni Paziente $P_i$ ha designato un Tutor fidato $T_i$, come un membro della famiglia o il responsabile della struttura in cui il paziente vive, che sarà responsabile della gestione delle chiavi di crittografia e dell'accesso alle entità che potrebbero avere bisogno di accedere ai dati. Queste entità, definite come Utenti ($U_i^j$), includono operatori, membri della famiglia, dispositivi di elaborazione dati e algoritmi per la gestione e il controllo del robot.
Gli Utenti ($U_i^j$) possono interrogare il $DB$ e recuperare i dati relativi al Paziente a loro collegato in base alle politiche di accesso dati definite dal Tutor per ciascun Utente. Potrebbero esserci uno o più Utenti $U_i^1, \ldots, U_i^j$ associati allo stesso Paziente $P_i$. Per semplificare la notazione, a volte ci riferiremo a $U_i^1$ come $U_i$.

Per garantire quanta più sicurezza possibile nelle comunicazioni e con lo scopo di evitare che degli attori malevoli si fingano nodi legittimi del sistema, utilizziamo uno strumento che in crittografia è definito come Public Key Certificate (certificate): esso è un documento elettronico utilizzato per dimostrare la validità di una chiave pubblica [81].
Nell'architettura, abbiamo anche deciso di utilizzare Mutual Transport Layer Security

(mTLS), una versione aggiornata di TLS in cui sia il client che il server devono essere autenticati per stabilire una connessione. Nella comunicazione mTLS, il certificate del Server assicura che la comunicazione tra un client e un server sia sicura. Nel frattempo, il certificato del client autentica il client che si connette a un servizio TLS.
Anche se questa è la soluzione preferita che può essere implementata in determinati casi, potrebbe essere difficile quando una delle parti non può essere dotata di un certificate. In tali casi, assumiamo che la connessione sia comunque sicura, anche se cerchiamo di farci affidamento il meno possibile.
In questa prospettiva, un metodo particolare per garantire che il mittente sia una parte legittima è il certificate pinning [82], in cui il destinatario di un messaggio fissa il certificate o il suo hash nel codice di configurazione.

Per garantire l'anonimato e l'impossibilità di rintracciare i dati, SSRA utilizza una tecnica di pseudonimizzazione. Pertanto, definiamo lo pseudonimo di un paziente $P_i$ come $\bar{P}_i$. Uno pseudonimo viene generato attraverso un Cryptographically Secure PseudoRandom Number Generator (CSPRNG) con un *seed* che deve essere casuale e ben protetto dal fornitore di servizi. In questo modo, si impedisce la creazione di un collegamento diretto tra il paziente e i suoi pseudonimi, con la possibilità di cambiare lo pseudonimo in qualsiasi momento generando un nuovo $\bar{P}_i'$ casuale e aggiornando le voci relative nel $DB$.

Per consentire un accesso dettagliato ai dati, la nostra architettura utilizza un tipo speciale di crittografia basata sugli attributi che definisce la politica di accesso all'interno del *ciphertext*. Questa è chiamata CP-ABE [83]. Con questo metodo, i blocchi di dati vengono crittografati con politiche attributo che le chiavi segrete devono rispettare per decifrare con successo.

Nella nostra architettura, definiamo due tipi di attributi. Uno è collegato al proprietario dei dati e, di conseguenza, al pseudonimo del paziente $\bar{P}_i$. L'altro si basa sulla restrizione di accesso che desideriamo imporre su ciascun tipo di dato. Definiamo questi attributi come $d_i$. Ogni utente $U_i$ dovrebbe avere accesso a uno o più tipi di dati in base al proprio modello di minacce. Questo schema crittografico fornisce una grande flessibilità, consentendo operazioni matematiche e logiche con operatori come AND (&) e/o OR (|) all'interno delle politiche. In SSRA, impieghiamo uno schema particolare per definire le politiche $\mathcal{P}$ per definire l'accesso dell'utente composto da una congiunzione del pseudonimo del paziente e una disgiunzione di tutti i tipi di dati che dovrebbero essere accessibili da quell'utente:

$$\mathcal{P} := \bar{P}_i \ \& \ (d_1 | \dots | d_j), \tag{4.1}$$

dove $\bar{P}_i$ rappresenta lo pseudonimo di un paziente, mentre $d_1 | \dots | d_j$ è una disgiunzione di tutti gli attributi dati disponibili per l'utente.
Durante la configurazione di questo metodo di crittografia, vengono generate due chiavi: la *master key* ($MK_i$) e la chiave pubblica $PK_i$. La prima è una chiave segreta di base che non possiede alcun attributo, ed è utilizzata per generare chiavi segrete. Dall'altra parte, $PK_i$ è utilizzata per crittografare i dati e dovrebbe essere associata agli attributi di crittografia. Entrambe le chiavi sono generate dal Tutor $T_i$, come spiegato nelle sezioni seguenti (Sottosezione 4.3.1). Tuttavia, $PK_i$ viene salvata sul Server per consentire la crittografia dei dati, mentre $MK_i$ è conservata in modo sicuro (possibilmente utilizzando un TPM) dal Tutor $T_i$. La funzione di crittografia $E$ è

definita come:

$$E(\mathcal{M}, \mathcal{P}, MK_i) = \mathcal{C}, \tag{4.2}$$

dove $\mathcal{M}$ è un blocco di dati da crittografare e $\mathcal{C}$ è il relativo *ciphertext*.

Successivamente, nel processo di generazione delle chiavi, una funzione $G$ viene utilizzata per generare le chiavi degli utenti. Essa richiede $MK_i$ insieme a una politica attributo $\mathcal{P}$:

$$G(MK_i, \mathcal{P}) = SK_{\mathcal{P}} = SK_{\bar{P}_i \ \& \ (d_1|...|d_j)}. \tag{4.3}$$

Per semplificare la notazione, a volte ometteremo gli attributi dati da $SK_{\bar{P}_i}$ quando è chiaro che la chiave concede l'accesso solo all'insieme di dati consentiti al particolare utente.

$SK_{\bar{P}_i}$ è quindi utilizzata dagli Utenti nella funzione di decrittazione $D$ per ottenere $\mathcal{M}$ da $\mathcal{C}$ come segue:

$$D(\mathcal{C}, SK_{\mathcal{P}}) = \mathcal{M}. \tag{4.4}$$

Per questo motivo, possiamo inoltre dire che:

$$\mathcal{M} = D(E(\mathcal{M}, \mathcal{P}, MK_i), SK_{\mathcal{P}}). \tag{4.5}$$

Nel seguito, analizziamo dettagliatamente le varie fasi della comunicazione nella nostra architettura, ovvero la configurazione, l'aggiunta di un nuovo utente, la comunicazione tra i nodi e la rigenerazione di chiavi e pseudonimi.

### 4.3.1 Setup

In questa sezione, spieghiamo i meccanismi di configurazione. L'instaurazione di un collegamento sicuro tra il Server e il Robot Sociale avviene *offline*, con uno scambio e il *pinning* dei certificate, come segue.

- Il Server invia il suo certificate ($CERT_S$) al Robot Sociale e lo fissa nel file di configurazione;
- Il Robot Sociale invia il suo certificate ($CERT_{R^j_{P_i}}$) al Server e lo fissa nel file di configurazione;
- I due nodi instaurano una connessione sicura utilizzando mTLS, che fornisce riservatezza, integrità e autenticazione [76].

Successivamente, un Tutor $T_i$ firma un contratto per aggiungere un paziente $P_i$ al sistema. Al Tutor viene fornito un nome utente $\bar{T}_i$ utilizzato per accedere al sistema e un pseudonimo per il paziente generato utilizzando un CSPRNG come $CSPRNG(seed) = \bar{P}_i$, che viene collegato dal Server a uno o più certificate dei robot $CERT_{R^j_{P_i}}$ assegnati al paziente. Questo processo viene effettuato *offline* o tramite canali sicuri che possono includere verifiche di identità e altre forme di convalida.
Successivamente, il resto degli scambi potrebbe avvenire attraverso una connessione TLS tra il Tutor $T_i$ e il Server $S$ come segue.

- Il Server invia $\bar{P}_i$ al Tutor $T_i$;
- Il Tutor genera le chiavi $MK_i$ e $PK_i$ collegate al paziente $\bar{P}_i$;

- Il Tutor invia $PK_i$ al Server.

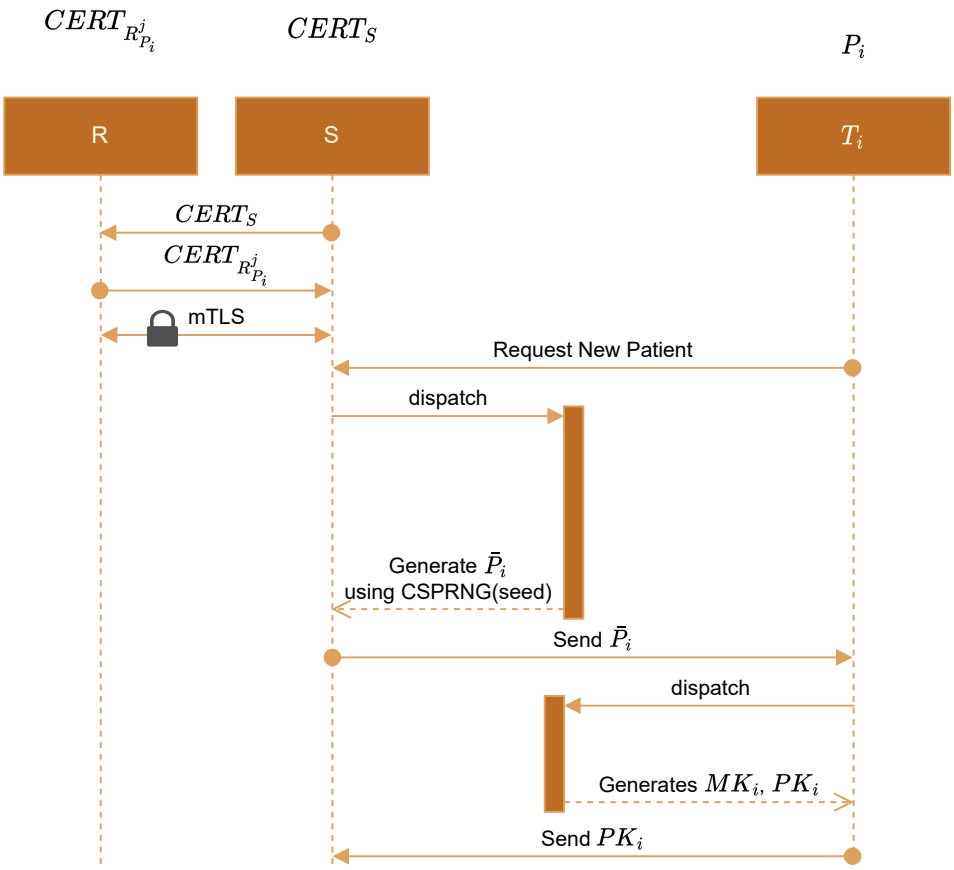


**Figura 4.4:** Operazioni di Setup

### 4.3.2 Aggiunta Utente

Il Tutor concede l'accesso a determinati tipi di dati relativi a un Paziente $P_i$ a uno o più Utenti $U_i^j$. A ciascun Utente può essere concesso un diverso insieme di attributi dati in base alle reali necessità, seguendo il principio del privilegio minimo [84]. Per consentire lo scambio di chiavi segrete attraverso un canale non sicuro tra due parti, SSRA sfrutta la fiducia nel Server per condividere i parametri per l'utilizzo di uno scambio chiave di Diffie-Hellman [85], come segue.

- Il Tutor $T_i$ apre una connessione TLS con il Server, inviando l'identità del nuovo Utente $U_i^j$;
- Il Server genera i parametri pubblici necessari per lo scambio chiave di Diffie-Hellman e risponde con essi;
- Il Server apre una connessione TLS con $U_i^j$ e invia i parametri pubblici;

- $T_i$ e $U_i^j$ utilizzano $p, g$ seguendo il protocollo di scambio chiave Diffie-Hellman [85] per condividere una chiave AES e aprire una connessione sicura tra le due parti;
- Il Tutor genera una $SK^j_{\bar{P}_i}$ basata sugli attributi $\mathcal{P}$ associati all'Utente $U_i^j$;
- Il Tutor invia $SK^j_{\bar{P}_i}$ a $U_i^j$ attraverso il canale sicuro.

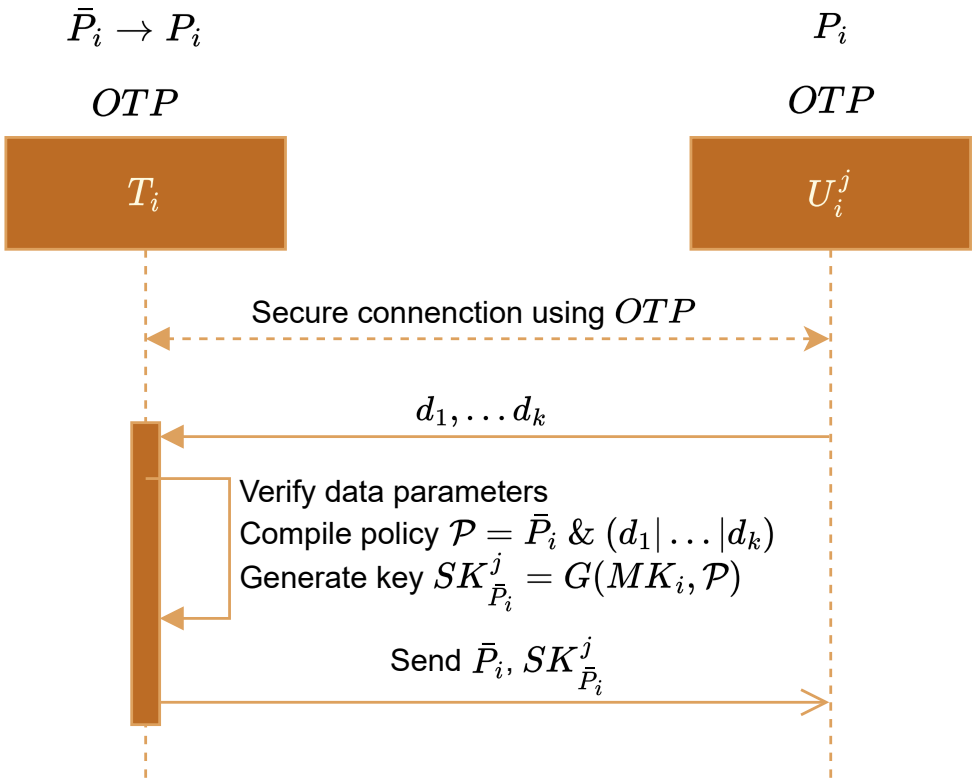


**Figura 4.5:** Operazioni di Aggiunta Utente

### 4.3.3 Operazioni Standard

Dopo la fase di configurazione, il Robot è pronto per raccogliere dati e seguire gli ordini del Server. I dati generati vengono inviati tramite mTLS al Server come segue.

- Un Robot $R_i^j$ genera un blocco di dati $\mathcal{M}$. Un timestamp[g] $t$ viene aggiunto al contenuto effettivo dei dati $\mathcal{M}$;
- Il Robot invia $[\mathcal{M}, t]$ a $S$ attraverso un canale mTLS;
- Il Server verifica che il messaggio ricevuto sia ancora valido verificando se $t \in (t_0, t_0 + \epsilon)$, dove $t_0$ è il tempo corrente e $\epsilon$ è un parametro di sicurezza. Se la verifica ha successo, il Server crittografa i dati $E(\mathcal{M}, \mathcal{P}, PK_i) = \mathcal{C}$, dove $\mathcal{P} = \bar{P}_i \ \& \ d_x$, dove $d_x$ definisce gli attributi del tipo di dati;

- Il Server memorizza il *ciphertext* $\mathcal{C}$ nel $DB$.

I dati memorizzati nel $DB$ possono essere richiesti dagli utenti in base ai loro privilegi e successivamente decifrati localmente utilizzando la chiave segreta di ciascun utente, come segue.

- Un Utente $U_i^j$ invia una richiesta di dati insieme ad alcuni timestamp che indicano quali dati vengono richiesti. In un'implementazione reale, è possibile implementare un'interfaccia web che richieda nome utente e password per facilitare l'interazione dell'utente con il sistema;
- Il Server recupera il *ciphertext* $\mathcal{C}$ richiesto relativo a $\bar{P}_i$ dal $DB$;
- Il Server invia $\mathcal{C}$ al richiedente $U_i^j$;
- $U_i^j$ decifra localmente i dati $D(\mathcal{C}, SK_{\bar{P}_i}) = \mathcal{M}$. La decrittazione fallisce se la chiave segreta dell'utente $SK_{\bar{P}_i}$ non ha gli attributi corretti per decifrare $\mathcal{C}$.

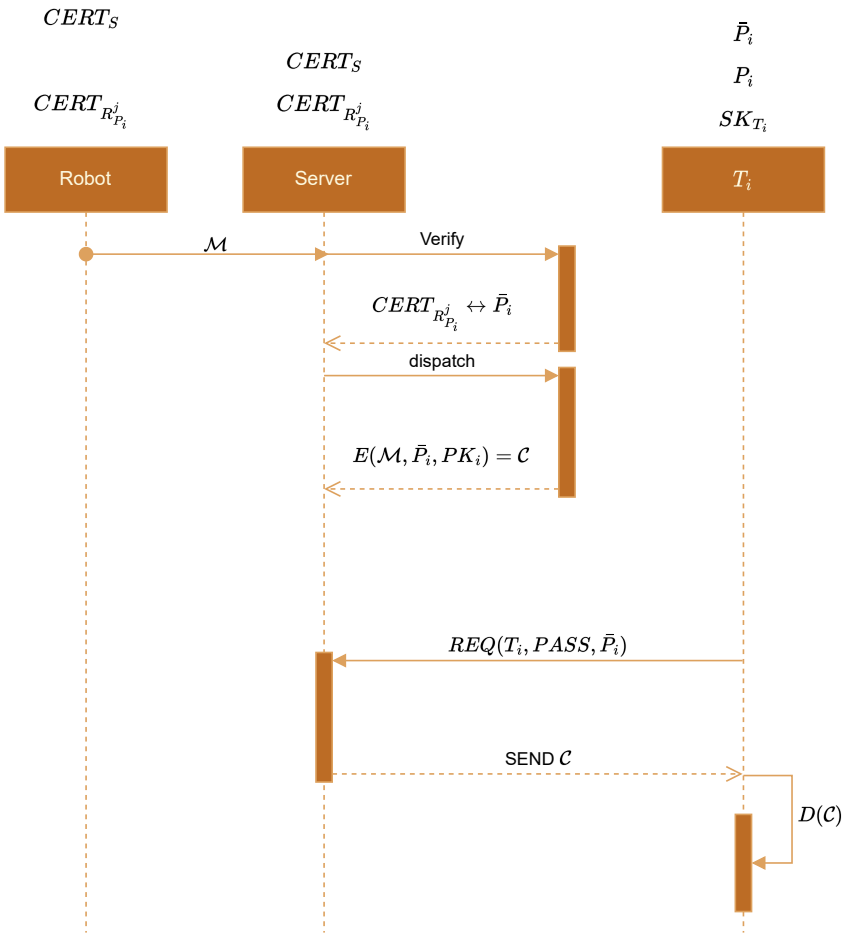


**Figura 4.6:** Operazioni Standard

### 4.3.4 Rigenerazione di Chiavi e Pseudonimi

Può accadere che, per qualsiasi motivo, una chiave segreta venga compromessa da un attaccante (ad esempio, infezione da malware su un laptop del cliente). In questi casi, l'architettura può reagire ed emettere nuove chiavi. Il processo è simile alla prima generazione di chiavi durante la configurazione (Sottosezione 4.3.1). Tuttavia, per garantire la segretezza in avanti, le vecchie voci nel $DB$ dovrebbero essere decifrate e crittografate nuovamente con la nuova chiave.

# Capitolo 5

# Scelte Implementative

*In questo capitolo si mostrerà come il prototipo della architettura è stato adattato per una realtà aziendale del settore.*

## 5.1 Descrizione dello Scenario

Lo scenario principale è rappresentato da un ente di sanità pubblica a livello regionale o nazionale in Italia, specificamente legato al Servizio Sanitario Nazionale (SSN) e alle Azienda Unità Locale Socio Sanitaria (AULSS). Queste entità hanno l'obiettivo ultimo di monitorare individui vulnerabili senza la necessità di mobilitare medici, infermieri e operatori socio-sanitari a meno che non vi sia una genuina necessità. Questo approccio consente un contatto continuo con gli individui assistiti attraverso il monitoraggio, questionari e videochiamate condotte da robot sociali e dispositivi mHealth.
In dettaglio, gli attori coinvolti sono:

- Paziente: l'individuo da monitorare, tipicamente una persona anziana autosufficiente con capacità motorie e sociali principalmente preservate;
- Autorità Sanitaria: rappresentata da un ente pubblico responsabile della sanità;
- Operatore: un individuo specializzato impiegato dall'Autorità Sanitaria il cui scopo è interfacciarsi tra il paziente e l'Autorità Sanitaria. Questa persona interagisce direttamente con il soggetto monitorato attraverso videochiamate e può accedere ai dati medici del paziente in qualsiasi momento per valutare progressi a breve e lungo termine;
- Terze Parti: rappresentate da individui con un interesse diretto nella situazione attuale del soggetto monitorato, come familiari, medici di medicina generale e, in alcuni casi, istituti di ricerca.

L'architettura assume che il robot sociale, posizionato nella residenza del paziente e fornito dall'Autorità Sanitaria, abbia una connessione internet stabile. Questo è cruciale poiché il suo comportamento è determinato da un albero decisionale situato su un server remoto, discusso successivamente in questo paragrafo. Se il robot non è connesso alla rete, le sue funzionalità saranno significativamente ridotte, rendendolo

essenzialmente un semplice automa a riposo.

I dati raccolti dal robot rientrano in tre categorie: analisi ambientale (audio e video) completati dai dati ricevuti da dispositivi indossabili (da dispositivi mHealth), telemetria e interazioni.
I dati provenienti dai flussi audio e video vengono analizzati da software di terze parti, mentre tutti gli altri dati vengono elaborati internamente.

Questa architettura mira a mantenere un registro paziente completo ed estremamente preciso, documentando eventi registrati dal primo giorno in cui il robot è posizionato nella residenza.
Gli obiettivi finali di questo progetto sono raccogliere un record clinico completo, garantire un accesso ai dati dettagliato e preservare la sicurezza dei dati raccolti in ogni fase del processo.

### 5.1.1 *Vivaldi*

*Vivaldi* è l'Intelligenza Artificiale programmata da Omitech Robot per rendere naturale l'interazione tra il robot e l'utente. Essa controlla e modifica i comportamenti dei robot reagendo in base alle informazioni e agli stimoli raccolti da vari dispositivi, tra cui i robot stessi.
Questa Intelligenza Artificiale rappresenta, nel progetto, l'albero decisionale da cui derivano i comportamenti del robot.

## 5.2 Adattamento dell'Architettura

Per rendere possibile lo studio di fattibilità per il caso proposto, è stato necessario ritoccare alcuni aspetti di SSRA.
Tra questi, si è reso necessario definire le responsabilita degli Operatori e delle Terze Parti. Per questo motivo, dopo un attento confronto con l'azienda, è stato deciso di assimilare le responsabilità degli Operatori a quelli del Tutor, mentre le Terze Parti sono state assimilate agli Utenti. Analizzando invece nel dettaglio il Server, è stato deciso di suddividerlo in vari moduli, la cui sottostruttura presenta:

- Server Cuscinetto:
  - Gestione delle comunicazioni Server $\leftrightarrow$ Robot;
  - Gestione delle comunicazioni Server $\leftrightarrow$ Operatore (Tutor) / Terze Parti (Utenti);
  - Gestione della comunicazione tra Operatore (Tutor) e Terze Parti (Utenti) nella aggiunta degli ultimi come Utenti del sistema;
- Access Control System:
  - Gestione dei diritti di accesso per le Terze Parti e gli Operatori;
- Modulo CP-ABE:
  - Crittazione dei blocchi dati;
  - Comunicazione con $DB$ per salvataggio e recupero dei blocchi dati crittati.

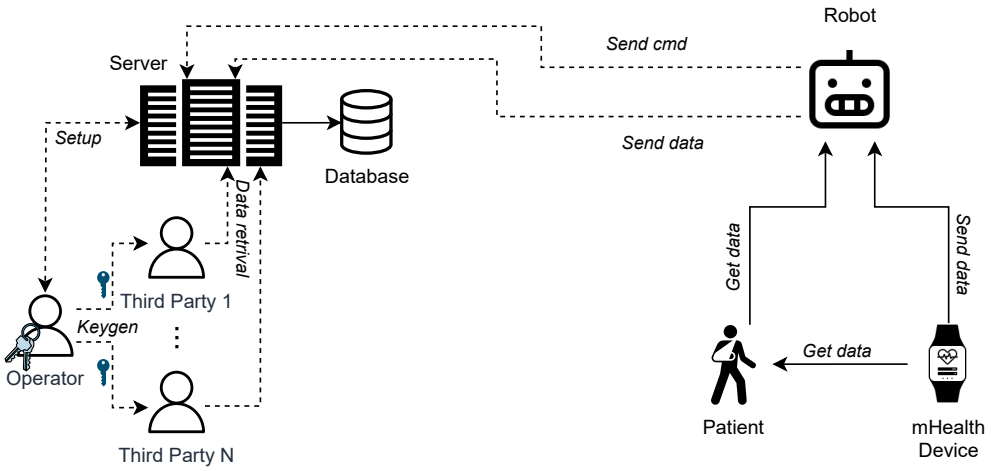


**Figura 5.1:** Diagramma di SSRA adattato

## 5.3 Implementazione dell'Architettura

In aggiunta a quanto riportato nella Sezione precedente (5.2), per poter completare lo studio di fattibilità, è stato necessario definire alcuni aspetti minori dell'architettura.

### 5.3.1 Risorse per l'Accesso al Sistema

Per poter accedere al sistema, sia gli Operatori che le Terze Parti devono essere in possesso di un account. Per questo motivo, è stato necessario definire un passaggio nel quale il gestore del servizio fornisse il necessario per l'accesso al sistema.
Gli elementi ritenuti necessari sono stati:

- Credenziali per l'accesso al sistema:
  - Username e password (quest'ultima generata casualmente);
- One-Time Password (OTP)[g] per la registrazione del paziente:
  - Da utilizzare al primo accesso per effettuare un collegamento diretto tra il Tutore e il Paziente.

### 5.3.2 Definizione di Processi

- Registrazione e primo accesso:
  - Processo che avviene da remoto utilizzando username, password e OTP. Al termine di questo processo viene richiesto il cambio della password generata casualmente;
- Accessi successivi al primo:
  - Processo che avviene da remoto utilizzando username e password;

- Aggiunta di una Terza Parte:
  - Processo che avviene da remoto, fare riferimento alla Sottosezione 4.3.2;
- Aggiornamento dei Robot Sociali:
  - Processo che può avvenire sia in remoto (tramite collegamento al Server attraverso canali sicuri) che *offline* avendo scaricato l'aggiornamento ufficiale da un canale sicuro su di un dispositivo di memoria esterna;
- Aggiornamento dei certificate per il *certificate pinning*:
  - Possibilmente *offline* avendo scaricato l'aggiornamento ufficiale da un canale sicuro su di un dispositivo di memoria esterna, alternativamente in remoto tramite il download del nuovo certificato ufficiale tramite un canale sicuro;
- Rigenerazione delle chiavi e dello pseudonimo:
  - Processo che avviene in remoto, fare riferimento alla Sottosezione 4.3.4.

### 5.3.3 Formalizzazione dei Flussi Dati

Considerando che i principali fruitori dei dati saranno *Vivaldi* (l'albero decisionale del robot sociale in Cloud Computing), l'Operatore (Tutor) e le Terze Parti (Utenti), è stato necessario definire i flussi dati tra i vari attori, sapendo che i dati disponibili sono:

- *Behavior Report*: collezione che contiene informazioni relative all'albero decisionale per l'istanza attuale. I dati sono aggiornati ciclicamente con una frequenza di aggiornamento o di un secondo o relativa ad eventi chiave;
- *Behavior User*: collezione che contiene la definizione degli utenti a cui fa riferimento un comportamento;
- *Maps*: collezione che contiene le informazioni relative alla mappa usata dal robot;
- *Status*: collezione che contiene informazioni relative allo stato e alle metriche del robot;
- *Records*: collezione che contiene le informazioni del sistema del robot e le funzionalità attive;
- *Routine*: collezione che indica dove potrebbe trovarsi l'utente in un determinato momento della giornata. Il calcolo è dinamico ed è basato sulla frequenza con la quale il paziente si trova in una determinata stanza in un determinato frangente temporale;
- *Survey* [WIP]: collezione che contiene le domande poste dal robot e le risposte date dal paziente ai questionari proposti dal sistema;
- *Body Detection* [WIP]: collezione che contiene informazioni relative al *body detection* che avviene tramite servizi terzi;
- *UIR* [WIP]: questi dati sono generati dal nucleo comportamentale che si occupa di rielaborare i dati raccolti durante il comportamento e faranno parte del profilo del paziente per la sua diagnostica;

- *Alert* [WIP]: collezione che contiene tutti i messaggi di allerta inviati all'operatore da und determinato paziente. La collezione contiene quindi gli eventi che hanno azionato l'*alert* e il tipo di messaggio inviato all'operatore (questo può essere testuale o multimediale).

Avendo quindi definito i flussi e le responsabilità degli attori, è stato possibile definire la suddivisione dei dati tra i vari attori nella Tabella 5.1 sotto presentata.

| | *Vivaldi* | Terza Parte | Operatore |
|---|---|---|---|
| Behavior Report | ● | ○ | ○ |
| Behavior User | ◐ | ○ | ○ |
| Maps | ● | ○ | ○ |
| Status | ● | ○ | ○ |
| Records | ● | ● | ● |
| Routine | ● | ○ | ◐ |
| Survey | ● | ◐ | ● |
| Body Detection | ● | ○ | ○ |
| UIR | ◐ | ○ | ● |
| Alert | ◐ | ● | ○ |

●= Accessibile, ◐= Da definire caso per caso, ○= Non Accessibile

**Tabella 5.1:** Tabella di accessibilità ai dati per i vari attori

### 5.3.4 Implementazione della Crittografia

Come precedentemente anticipato, un modulo del server è quello dedicato alla cifratura. Per dimostrare il funzionamento di tale è stato ricercato online una versione *open source* di un modulo CP-ABE.
La scelta, data anche dalla scarsissima presenza di alternative funzionanti senza modifiche maggiori, è ricaduta su *RABE* [86], ossia un modulo Rust[g] che permette di implementare velocemente la ABE, tra cui anche la CP-ABE.

Dopo aver trovato il modulo, è stato necessario adattarlo alle esigenze del progetto. Per questo sono stati creati due file *.sh* per testare le funzionalità e la rapidità del modulo su due (2) algoritmi: BSW [83] e AC17CP [87].

I due file sopra citati, denominati rispettivamente `poc_BSW.sh` e `poc_AC17CP.sh`, sono essenzialmente identici, l'unica differenza risiede nella specifica dell'algoritmo da utilizzare per la cifratura dei dati e per la creazione delle chiavi.
Questi due sono suddivisibili in quattro (4) operazioni fondamentali, a partire dal setup:

```
# Il codice sottostante genera la master key e la public key
./rabe --s ALGORITHM setup
# La funzione setup utilizzata per la creazione della msk e della
    pk
# --s ALGORITHM indica l'algoritmo da utilizzare per la cifratura
    (BSW o AC17CP)
```

**Listing 5.1:** Operazione di Setup

Successivamente, è necessario generare le chiavi per gli utenti:

```
./rabe --s AC17CP keygen --a 'Attr1 Attr2'
# La funzione keygen viene utilizzata per la creazione delle sk
# --a 'Attr1 Attr2' indica gli attributi presentati dalla sk
```

**Listing 5.2:** Operazione di Keygen

Infine, è possibile cifrare i dati:

```
./rabe --s AC17CP --l HUMAN encrypt File.ext '"Attr1" "Attr2"'
# La funzione encrypt viene utilizzata per la cifratura dei dati
# Il file File.ext contiene i da cifrare
# La stringa '"Attr1" "Attr2"' indica gli attributi necessari per
    decifrare il file
```

**Listing 5.3:** Operazione di Cifratura dei Dati

Arrivati a questo punto, il modulo genererà un secondo file denominato `File.ext.ct` che conterrà i dati cifrati.
Per decifrare i dati, è necessario utilizzare la funzione *decrypt*:

```
./rabe --s AC17CP decrypt File.ext.ct sk.key
# La funzione decrypt viene utilizzata per la decifratura dei
   dati
# Il file File.ext.ct contiene i da cifrare
# Il file sk.key contiene la chiave segreta necessaria per
   decifrare il file
```

**Listing 5.4:** Operazione di Decifratura dei Dati

Il file decrittato, per come è stato implementato il modulo, riscriverà per intero il file originale (`File.ext`), di fatto sovrascrivendo i dati iniziali.

Per valutare l'efficacia del modulo, sono stati testati i tempi necessari per la cifratura e la successiva decifratura di un file di testo di `5MB` contenente 466.500 parole estrapolate dal dizionario inglese[88] utilizzando la CP-ABE, RSA e AES. I tempi rilevati sono stati i seguenti:

| | Encrypt | Decrypt | Totale |
|---|---|---|---|
| **AES** | $\approx 0.505ms$ | $\approx 0.485ms$ | $\approx 0.990ms$ |
| **RSA** | $\approx 18500ms$ | $\approx 5500ms$ | $\approx 24000ms$ |
| **CP-ABE - BSW** [83] | $\approx 0.016ms$ | $\approx 0.015ms$ | $\approx 0.031ms$ |
| **CP-ABE - AC17CP** [87] | $\approx 0.027ms$ | $\approx 0.028ms$ | $\approx 0.055ms$ |

**Tabella 5.2:** Tabella di Confronto dei Tempi

Sorvolando sui vantaggi dovuti alla granularità di accesso offerto da CP-ABE, si può comunque notare che la CP-ABE risulta estremamente più veloce rispetto a RSA e AES, garantendo inoltre la possibilità di cifrare blocchi dati di qualsiasi dimensione e tipo in maniera facile e veloce.

# Capitolo 6

# Analisi della Sicurezza

*In questo capitolo verranno discussi i metodi utilizzati per prevenire le minacce relative ai Security Requirements descritti nella Sezione 4.2.*

## 6.1 Analisi della Sicurezza

**R1: Integrità dei Dati** L'integrità della comunicazione è garantita in ogni collegamento. Tra $R_i^j$ e il Server, mTLS assicura l'integrità utilizzando un algoritmo di hash per calcolare un digest del messaggio. Anche la trasmissione dei dati dal Server a $U_i$ è protetta, poiché la comunicazione avviene tramite TLS.

**R2: Confidenzialità dei Dati.** La confidenzialità dei dati è una delle richieste più importanti quando si tratta di dati medici. mTLS garantisce la confidenzialità del collegamento tra $R_i$ e $S$, mentre CP-ABE assicura che i dati intercettati tra $S$ e $U_i$ siano sempre criptati. Infatti, solo un destinatario dei dati che ha accesso a una chiave segreta con l'attributo corretto è in grado di decifrare un messaggio. Un $U_i$ curioso o compromesso potrebbe richiedere i dati di un paziente $\bar{P}_z$ non associato a lui. Tuttavia, poiché $SK_{\bar{P}_z \& d_x}$ (noto a $U_z$) non contiene attributi relativi a $\bar{P}_z$, non è in grado di decifrare quel cifrato. La confidenzialità dei dati è mantenuta anche per il server, che non conosce alcuna chiave segreta $SK$. Anche durante la generazione delle chiavi segrete da $T_i$ per ogni $U_i^j$, il Server agirà solo come facilitatore per consentire uno scambio sicuro e privato tra le due parti.

**R3: Controllo dei Danneggiamenti** In ogni momento, un nodo compromesso non può danneggiare un altro o più componenti della rete. Il Social Robot non sarebbe in grado di danneggiare un essere umano perché l'unica fonte valida per ricevere istruzioni è il server attraverso una connessione sicura. Considerando che il robot lavora con una strategia decisionale remota (potrebbe essere un albero decisionale o un algoritmo più avanzato basato su ML [3]), se compromesso e disconnesso da Internet, diventerebbe solo un automa inattivo. Inoltre, nessun altro nodo nella rete è autorizzato a inviare istruzioni ai robot. Un utente compromesso ha solo la capacità di leggere il suo sottoinsieme di dati relativi al suo paziente associato, considerando che la sua vera identità è nascosta dietro lo pseudonimo. Poiché viene fornito l'accesso in sola lettura, la modifica malintenzionata di qualsiasi informazione non è possibile. Un

Tutor compromesso acquisisce la capacità di leggere i dati del suo paziente associato, aggiungere un nuovo utente e generare un nuovo set di chiavi, ma tali comportamenti non possono danneggiare altri nodi. Tuttavia, un Tutor compromesso potrebbe aggiungere un nuovo utente alla rete, consentendo a un utente malintenzionato di leggere i dati del paziente monitorato. Sebbene ciò sia intrinseco alle capacità del Tutor, un Tutor compromesso non può emettere chiavi per i dati dei pazienti non sotto il suo controllo.

**R4: Resistenza all'Impersonificazione** Per un attaccante è impossibile impersonare un nodo nella rete nella comunicazione con gli altri. Nel collegamento tra $R_i$ e $S$, ciò è garantito dal pinning del certificato di $CERT_{R_i}$ e $CERT_S$. Poiché il robot fissa $CERT_S$ nel suo file di configurazione, ogni volta che qualcuno cerca di impersonare $S$, il certificato non corrisponderà a quello fissato. D'altra parte, $S$ mantiene un elenco di tutti i certificati dei robot nel suo $DB$. Pertanto, se un robot $R_i$ cerca di comunicare con $S$, esso cercherà nel $DB$ un certificato corrispondente $CERT_{R_i}$. Se nessun certificato viene trovato, la comunicazione sarà interrotta. D'altra parte, un $U_i$ curioso potrebbe impersonare un altro utente $U_z$ e richiedere alcuni dati $\mathcal{C}$ appartenenti a $\bar{P}_z$. Tuttavia, non sarà possibile per $U_i$ decifrare $\mathcal{C}$ poiché $U_i$ non conosce alcuna chiave $SK_{\bar{P}_z}$ contenente l'attributo $\bar{P}_z$, necessario per decifrare $\mathcal{C}$.

**R5: Continuità** L'obiettivo della rete è fornire un servizio continuo. Tuttavia, di solito non è possibile ed efficiente dal punto di vista dei costi sviluppare e implementare reti ad-hoc presso ogni domicilio del paziente. Pertanto, sono generalmente impiegate connessioni domestiche standard come il WiFi. Questo approccio potrebbe avere un impatto sulla affidabilità della connessione, che potrebbe subire occasionali interruzioni del servizio a causa di difetti nel sistema o minacce portate avanti da malintenzionati, come per esempio attacchi Denial of Service (DoS)[g] e/o Distributed Denial of Service (DDos)[g]. Per evitare che i componenti principali (il Robot e il Server) si disconnettano, esistono alcuni metodi, come il Deep Packet Inspection (DPI) da parte del Server (anche se computazionalmente difficile su grandi quantità di dati o in elaboratori con bassa potenza computazionale), la stesura di una blacklist e whitelist per gli indirizzi IP e la riduzione della superficie di attacco (ossia, esporre al mondo esterno meno servizi possibili). Un approccio valido è anche quella di predisporre impiegare una connessione secondaria per il Robot, come un modem Long Term Evolution (LTE)[g] o 5G[g], da utilizzare come connessione di backup quando le altre falliscono. Inoltre, poiché il robot non ha abbastanza capacità di calcolo per eseguire modelli di ML, quando è scollegato dalla rete, concluderà le sue attività e entrerà in una fase di inattività, senza intraprendere altre azioni rischiose o di qualsiasi natura.

**R6: Assenza di Tracciabilità** L'assenza di tracciabilità del paziente è garantita perché solo il Tutor $T_i$ può collegare l'identità del paziente $P_i$ al suo pseudonimo $\bar{P}_i$. Tuttavia, se la rete viene a conoscenza di una chiave compromessa o di uno pseudonimo compromesso, $S$ genererà un nuovo pseudonimo e informerà $T_i$ dell'avvenuto, il quale procederà quanto prima alla generazione di un nuovo set di chiavi. Il nuovo pseudonimo sarà poi collegato al certificato del Robot ($CERT_{R_i}$).

**R7: Anti Replay Attack** Considerando che la connessione tra il Robot e il Server non avviene in tempo reale, i blocchi di dati vengono trasmessi periodicamente. Ciascun pacchetto include un timestamp $t$ crittografato e firmato, che il Server verifica per decidere se accettare o rifiutare il messaggio. La finestra di accettazione di un pacchetto

è definita da un parametro di sicurezza $\epsilon$, che rappresenta il periodo massimo entro il quale un attaccante deve reiterare l'invio del pacchetto. La definizione dei parametri di sicurezza dovrebbe considerare non solo l'aspetto della sicurezza, migliorato da un $\epsilon$ ridotto, ma anche l'usabilità, poiché i ritardi nella comunicazione dovrebbero invalidare il minor numero possibile di pacchetti, bilanciando così efficacemente sicurezza e praticità.

## 6.2 Confronto

Definito il soddisfacimento dei *Security Requirements* da parte di SSRA, è possibile confrontare nella Tabella 6.1 tale architettura con le altre prese di riferimento. SSRA

| | SR Based* | R1 | R2 | R3 | R4 | R5 | R6 | R7 |
|---|---|---|---|---|---|---|---|---|
| SSRA | ✓ | ● | ● | ● | ● | ● | ● | ● |
| Huang et al. [64] | ⊠ | ● | ● | ○ | ◐ | ○ | ● | ● |
| Tomaz et al. [65] | ⊠ | ○ | ● | ○ | ◐ | ○ | ● | ◐ |
| Loza et al. [59] | ✓ | ○ | ○ | ○ | ○ | ● | ○ | ○ |
| Bonaccorsi et al. [58] | ✓ | ○ | ○ | ○ | ○ | ○ | ○ | ○ |

*SR Based = Architettura esplicitata formalmente per social robot. ✓= Architettura per social robot. ⊠= Architettura non per social robot.

●= Soddisfatto, ◐= Parzialmente Soddisfatto, ○= Non Soddisfatto

**Tabella 6.1:** Requisiti di sicurezza soddisfatti da SSRA e nei principali lavori simili nella letteratura.

risulta essere quindi la prima architettura che considera esplicitamente la sicurezza e la privacy durante la progettazione di un'architettura di rete per robot sociali. Alcuni articoli propongono architetture di rete simili [58], [59], ma senza indagare sulla sicurezza dell'intero sistema. D'altro canto, altri lavori hanno affrontato argomenti simili in modo più ampio, considerando dispositivi medici generici [64] o sistemi di salute mobile [65]. In questa sezione, forniamo un confronto con i lavori correlati presenti nella letteratura.

Huang et al. [64] hanno proposto un'architettura di sicurezza generale legata ai dati medici. L'adattamento diretto al contesto dei robot sociali è complicato per diverse ragioni. Il modo in cui i pazienti ottengono i loro dati non è mai citato e costituisce un requisito fondamentale nel nostro scenario. Inoltre, l'architettura proposta non è adatta per gestire lo stato in tempo reale del paziente, come fanno continuamente i robot sociali.

Rispetto ad altri lavori [64], [65], proponiamo un'architettura che non utilizza una blockchain. Anche se molte soluzioni esistono per il suo utilizzo nella preservazione della privacy, spesso risulta non ancora sufficientemente regolamentata nei varei paesi, il che potrebbe rappresentare una limitazione per i fornitori di servizi.

Inoltre, in entrambi gli articoli [64], [65], R4 è solo parzialmente soddisfatto, poiché non viene considerato un attacco dell'evil twin. Infatti, un attore malintenzionato potrebbe installare un dispositivo non attendibile nella rete, impersonando un dispositivo legittimo.

Inoltre, Tomaz et al. [65] non soddisfa completamente R7. Questo perché i blocchi dati non includono un timestamp o un altro identificatore unico. Invece, viene utilizzato un identificatore incrementale, il che rende possibile lanciare attacchi di ripetizione in determinati contesti.

Altri articoli sono invece più focalizzati sui robot sociali, mentre gli aspetti di sicurezza hanno ottenuto poca o nessuna attenzione. L'architettura formalizzata da Loza et al. [59] considera ed implementa solo uno dei nostri requisiti, ovvero R5. Il requisito di continuità è garantito mediante l'uso di un server locale e di uno remoto. Il robot sociale è dotato di conoscenze essenziali sul suo comportamento previsto. Quindi, il server locale fornisce un piccolo sottoinsieme di azioni, mentre il server remoto fornisce l'intero set di capacità. In questa architettura, anche se il server locale viene disconnesso dal server remoto, il robot mantiene un certo range di capacità utile al paziente. Alcune delle soluzioni proposte possono essere implementate per aggiungere sicurezza e privacy. Ad esempio, l'implementazione di TLS può garantire l'integrità (R1) e la riservatezza (R2). La separazione dei ruoli e delle responsabilità degli utenti limita i movimenti laterali (R3), mentre l'implementazione del pinning del certificato garantisce la resistenza all'impersonificazione (R4). Infine, la pseudonimizzazione dovrebbe essere inclusa per garantire l'irrintracciabilità (R6), mentre identificatori univoci su ogni blocco impediranno gli attacchi di ripetizione (R7).
D'altra parte, l'architettura formalizzata da Bonaccorsi et al. [58] non cita né implementa alcuno dei nostri requisiti proposti. Tuttavia, similmente a Loza et al. [59], potrebbe essere adattata per supportare i requisiti di sicurezza presentati in questo lavoro.

## 6.3 Discussione

Abbiamo modellato il nostro sistema considerando il server parzialmente affidabile (in gergo, *semi-trusted*, cioè non in grado di leggere i dati prima della cifratura). Come spiegato, questa assunzione sembra ragionevole per alleviare il carico sui robot evitando la cifratura locale dei file. Tuttavia, potrebbero esserci contesti in cui il server è completamente inaffidabile (in gergo, *untrusted*), e sono comunque necessari gli stessi requisiti di sicurezza. In questo scenario, i dati devono essere ricevuti dal server già cifrati. Una soluzione potrebbe essere potenziare le capacità computazionali del robot e eseguire la cifratura direttamente all'interno del dispositivo. Tuttavia, ciò non è sempre pratico perché i) il robot dovrebbe essere il meno pesante possibile per evitare pericoli per i pazienti [12], e ii) il costo del robot dovrebbe essere mantenuto al minimo per massimizzare la copertura dei pazienti e essere competitivo rispetto all'assistenza umana. Una possibile soluzione alternativa, specialmente in strutture con molti robot e pazienti, potrebbe essere l'utilizzo di un'unità computazionale esterna da utilizzare come *gateway* per tutti i robot. Dovrebbe essere responsabile della cifratura dei dati e dell'invio al server. L'applicazione di queste soluzioni aumenta il costo del sistema ma riduce la parziale affidabilità del server. Ogni scenario di implementazione dovrebbe eseguire un'analisi della valutazione del rischio e definire l'architettura di conseguenza. Tuttavia, è importante notare che anche se il server ha accesso ai dati non cifrati, non sarà mai in grado di collegare lo pseudonimo associato ai dati con l'identità reale del paziente, che è nota solo al Tutor legittimo e agli utenti collegati e approvati dal Tutor stesso.

# Capitolo 7

# Conclusioni

## 7.1 Raggiungimento degli Obiettivi

Al termine del tirocinio, tutti gli obiettivi fissati nella Tabella 2.3 sono stati raggiunti, come riportato nella Tabella 7.1. Per quanto riguarda invece il tempo preventivato e

| ID | Descrizione | Soddisfatto? |
|---|---|---|
| RO-01 | Studio del background della verifica formale dei flussi di dati | ⊠ |
| RO-02 | Studio del background dei dispositivi medici robotici domiciliari e dati gestiti | ● |
| RO-03 | Studio del background dell'architettura dei robot sociali nella letteratura | ● |
| RO-04 | Studio del background dell'architettura generale | ⊠ |
| RO-05 | Progettazione di una architettura sicura generica | ● |
| RO-06 | Implementazione di una architettura sicura generica | ● |
| RO-07 | Verifica dell'architettura sicura generica | ⊠ |
| RO-08 | Studio del background delle blockchain e degli *smart contract* | ● |
| RO-09 | Studio del background della privacy e della crittografia | ● |
| RD-01 | Analisi dei flussi derivanti dalla architettura preesistente | ● |
| RD-02 | Adattamento dell'architettura sicura generica all'architettura preesistente | ⊠ |
| RF-01 | Formalizzazione dei flussi di dati generati dalla architettura preesistente | ● |
| RF-02 | Stesura del paper di ricerca | ● |

●= Soddisfatto, ◐= Parzialmente Soddisfatto, ○= Non Soddisfatto, ⊠= Rimosso

**Tabella 7.1:** Tabella degli Obiettivi con Relativo Indicatore di Soddisfacimento

quello effettivamente utilizzato, in riferimento alla Tabella 2.2, di seguito la suddivisione oraria:

| Descrizione dell'Attività | Ore Preventivate | Ore Effettive |
|---|---|---|
| **Background** | **60** | **85** |
| *Studio dei dispositivi medici robotici domiciliari* | 15 | 9 |
| *Studio dell'architettura dei robot sociali nella letteratura* | 15 | 32 |
| *Studio delle blockchain e degli smart contract* | 15 | 15 |
| *Studio della privacy e della crittografia* | 15 | 28 |
| **Studio della Architettura Proposta** | **30** | **35** |
| *Studio della architettura proposta* | 25 | 30 |
| *Studio della formalizzazione dei flussi di dati* | 5 | 5 |
| **Progettazione di una Architettura Sicura Generica** | **130** | **120** |
| **Adattamento dell'Architettura Sicura Generica all'Architettura Proposta** | **60** | **25** |
| **Studio di Fattibilità per l'Architettura Proposta** | **20** | **30** |
| **Stesura Paper** | **20** | **25** |

**Tabella 7.2:** Tabella delle Ore Preventivate ed Effettive

## 7.2 Conoscenze Acquisite

Durante il periodo di tirocinio, ho avuto l'opportunità di immergermi nel mondo della robotica, un campo che fino a quel momento mi era pressoché sconosciuto. Questa esperienza si è rivelata estremamente istruttiva, permettendomi di acquisire conoscenze approfondite riguardo alle applicazioni della robotica nel contesto sociale. In particolare, ho dedicato tempo ed energie allo studio dettagliato delle complesse architetture informatiche che sostengono il funzionamento dei robot sociali.

Ciò che ha suscitato il mio interesse in modo particolare è stata la componente della sicurezza informatica, un argomento che ha sempre occupato un posto di rilievo nei miei interessi non solo accademici, ma anche personali. Nel contesto della robotica sociale, ho avuto l'opportunità di esplorare le sfide uniche legate alla sicurezza, considerando scenari in cui nodi e collegamenti sono esposti a potenziali minacce, soprattutto in un contesto dove i dati sono estremamente sensibili, come nel campo sanitario e assistenziale. La mia attenzione si è concentrata su come garantire l'integrità, la riservatezza e la disponibilità delle informazioni scambiate all'interno di questi sistemi, tenendo conto di possibili attacchi, come la compromissione dei nodi o il blocco delle connessioni.

Questo tirocinio ha arricchito la mia comprensione non solo dell'assai complesso mondo della robotica, ma ha anche ampliato la mia prospettiva sulla sicurezza informatica in contesti innovativi. Sono entusiasta di applicare queste nuove competenze nel mio percorso accademico, professionale e personale, evidenziando come l'integrazione tra robotica e sicurezza informatica possa svolgere un ruolo cruciale nello sviluppo di soluzioni tecnologiche avanzate e sicure per la società per gli anni a venire.

## 7.3 Il Paper IEEE

Terminata sia la formalizzazione dell'architettura che la sua implementazione per l'azienda tramite uno studio di fattibilità, insieme al Dott. Donadel e con la supervisione del Prof. Conti abbiamo iniziato la stesura di un *paper* intitolato *SSRA: A Secure Social Robot Architecture* con l'obiettivo di inviarlo per la pubblicazione al *Journal of Biomedical & Health Informatics (J-BHI)*, facente parte della Institute of Electrical and Electronics Engineers (IEEE)[g].

## 7.4 Valutazione Personale

Al termine delle ore di tirocinio non posso che affermare che sia stata una esperienza estremamente formativa che ha rappresentato un capitolo fondamentale del mio percorso accademico, offrendomi un'esperienza ricca di apprendimento. Durante questo periodo, ho avuto il privilegio di immergermi nelle complessità e nelle sfide legate alla progettazione di sistemi intelligenti per l'assistenza, sviluppando competenze e conoscenze che hanno contribuito in modo significativo alla mia crescita professionale e al mio bagaglio accademico.

La possibilità di approfondire la teoria di campi inizialmente sconosciuti, come quella dei social robot, della blockchain e della crittografia avanzata, è stata un aspetto distintivo di questo tirocinio. La formalizzazione dell'architettura per robot sociali ha richiesto una comprensione approfondita di concetti legati non solo alla robotica stessa, ma anche alle relazioni che intercorrono tra la persona e il robot e sul trattamento di dati sensibili come quelli sanitari. Questo processo di approfondimento teorico non solo ha consolidato le mie basi accademiche, ma ha anche fornito un solido fondamento per affrontare sfide pratiche nel mondo del lavoro.

La collaborazione con un'azienda del settore è stata un elemento chiave dell'esperienza di tirocinio, poiché ha permesso di collegare la teoria alla pratica. Lo studio di fattibilità portato nel contesto aziendale ha offerto un caso di studio adatto per provare a rendere concreto il lavoro che fino a quel momento era stato esclusivamente teorico. La possibilità di lavorare a stretto contatto con professionisti del settore ha rappresentato un valore aggiunto, consentendomi di cogliere sfumature e sfide che vanno al di là del contesto accademico. In conclusione, non posso che valutare più che positivamente il percorso di stage interno all'Università degli Studi di Padova offerto dal Prof. Conti e dal Dott. Donadel.

# Acronimi e abbreviazioni

**3DES** Triple Data Encryption Standard. 9

**ABE** Attribute-Based Encryption. 10, 35

**AES** Advanced Encryption Standard. 9, 28, 36

**AULSS** Azienda Unità Locale Socio Sanitaria. 31

**certificate** Public Key Certificate. 24–26, 34

**COVID-19** COrona VIrus Disease 2019. 19

**CP-ABE** Ciphertext-Policy Attribute-Based Encryption. 10, 11, 25, 32, 35–37

**CSPRNG** Cryptographically Secure PseudoRandom Number Generator. 25, 26

**DDos** Distributed Denial of Service. 38, 46

**DES** Data Encryption Standard. 9

**DoS** Denial of Service. 38, 46

**DPI** Deep Packet Inspection. 38

**ECC** Elliptic Curve Cryptography. 9, 10

**HCI** Human-Computer Interaction. 5

**HRI** Human-Robot Interaction. 4, 19

**IEEE** Institute of Electrical and Electronics Engineers. 43, 47

**IoT** Internet of Things. 6, 7

**J-BHI** Journal of Biomedical & Health Informatics. 43

**KP-ABE** Key-Policy Attribute-Based Encryption. 10

**LTE** glslinkltegLong Term Evolution. 38

**mHealth** Mobile Health. 7, 21, 24, 31, 32

**MITM** Man In The Middle. 21, 47

**ML** Machine Learning. 6, 37, 38

**mTLS** Mutual Transport Layer Security. 24–26, 28, 37

**NIST** National Institute of Standards and Technology. 21, 47

**OTP** One-Time Password. 33, 47

**PoC** Proof of Concept. 1, 17, 18, 47

**RC4** Rivest Cipher 4. 9

**RSA** Rivest–Shamir–Adleman. 9, 10, 36

**S2T** Speech-to-Text. 5, 6

**SSL** Secure Sockets Layer. 8

**SSN** Servizio Sanitario Nazionale. 31

**SSRA** Secure Social Robot Architecture. 23, 25, 27, 32, 39

**T2S** Text-to-Speech. 5, 6

**TLS** Transport Layer Security. 8, 21, 25–27, 37, 40

**TPM** Trusted Platform Module. 21, 25, 48

# Glossario

**5G** traducibile come *quinta generazione*, è uno standard di comunicazione wireless di quinta generazione che consente di trasmettere dati ad alta velocità per smartphone, tablet, laptop e dispositivi intelligenti. 38

**Big Data** traducibile come *dati di grandi dimensioni*, indica genericamente una raccolta di dati talmente estesa in termini di volume e varietà da richiede metodi analitici *ad hoc* per l'estrazione di informazioni utili. 8

**Blockchain** si riferisce ad una tecnologia di registri distribuiti che consente di registrare transazioni in modo sicuro, trasparente e immutabile. I dati archiviati sulla blockchain sono distribuiti su una rete di computer, il che rende difficile la loro modifica o falsificazione. 13, 14, 41–43

**Cloud Computing** traducibile come *elaborazione in cloud*, si riferisce alla fornitura di servizi di elaborazione, inclusi server, archiviazione, database, rete, software, analisi e intelligenza, su Internet ("il cloud") per offrire innovazione più rapida, risorse flessibili ed economie di scala. 6, 19, 20, 34

**DDoS** traducibile come *attacco distribuito di negazione del servizio*, rappresena una evoluzione del DoS dove il traffico di dei dati in entrata che inonda la vittima proviente da molteplici fonti diverse. 44

**DNS Cache Poisoning** traducibile come *avvelenamento della cache DNS*, è un attacco informatico in cui i dati vengono introdotti nella cache DNS di un server. Ciò può portare a un risultato indesiderato, ad esempio, se l'attaccante inserisce un indirizzo IP dannoso per un nome di dominio, il server DNS restituirà l'indirizzo IP dannoso quando viene richiesto il nome di dominio. 21

**DoS** traducibile come *negazione del servizio*, è un tipo di attacco informatico nel quale si fanno esaurire deliberatamente le risorse di un sistema informatico che fornisce un servizio ai *client*, ad esempio un sito web su un web server, fino a renderlo non più in grado di erogare il servizio ai *client* richiedenti. 44

**Hashing** si riferisce al processo di conversione di una stringa di input in una stringa di lunghezza fissa che rappresenta l'input originale. Un buon algoritmo di hashing ha le seguenti proprietà: è facile calcolare l'hash per qualsiasi dato, è difficile calcolare un messaggio che ha un hash specifico, è difficile modificare un messaggio senza cambiare il suo hash, è difficile trovare due messaggi diversi con lo stesso hash. 12

**IEEE** associazione internazionale di scienziati professionisti con l'obiettivo della promozione delle scienze tecnologiche. Tra gli scopi principali ci sono quelli di cercare nuove applicazioni e teorie nella scienza elettrotecnica, elettronica, informatica, biomedica e delle telecomunicazioni; a questo scopo organizza conferenze e dibattiti tecnici in tutto il mondo, pubblica testi tecnici e sostiene programmi educativi. Inoltre, si occupa di pubblicare standard per i campi prima citati. 44

**Jamming Attack** traducibile come *attacco di disturbo*, è un attacco informatico in cui un dispositivo di comunicazione viene reso inutilizzabile o meno efficiente. Ciò può essere fatto inviando segnali elettromagnetici che interferiscono con il segnale originale. 21

**MITM** traducibile come *uomo nel mezzo*, è un attacco informatico in cui un attaccante intercetta la comunicazione tra due parti e la controlla. L'attaccante può quindi inviare e ricevere messaggi che sembrano provenire da entrambe le parti. 45

**NIST** traducibile come *istituto nazionale di standard e tecnologia*, è un'agenzia federale statunitense che sviluppa e promuove le misure di sicurezza informatica e le raccomandazioni per gli Stati Uniti. 45

**Offband** si riferisce a un metodo di trasmissione delle informazioni che avviene su un canale separato rispetto a quello principale utilizzato per la comunicazione normale. Nel contesto della sicurezza informatica e della gestione delle reti, la comunicazione off-band viene utilizzata per scambiare informazioni critiche o sensibili in modo separato dalla rete principale. Questo approccio è progettato per ridurre il rischio di intercettazioni e attacchi da parte di terze parti malevole. 21

**OTP** traducibile come *password monouso*, è una password che viene utilizzata solo una volta e non può essere riutilizzata. Le password monouso possono essere utilizzate per autenticare un utente solo una volta, quindi anche se un utente dovesse rivelare accidentalmente la password, non potrebbe essere utilizzata da un attaccante per accedere al sistema. 45

**PoC** traducibile in italiano come *prova di fattibilità*, si riferisce a una realizzazione completa o abbozzata di un determinato progetto o metodo. Il suo scopo principale è quello di dimostrare la fattibilità o confermare la validità di alcuni prototipi o concetti fondamentali. 45

**Rumore Probabilistico** si riferisce a un rumore che può essere descritto da una distribuzione di probabilità. In questo caso, il rumore è descritto da una distribuzione di probabilità di Gauss. 9

**Rust** Linguaggio di programmazione di sistemi che si concentra sulla velocità, sulla sicurezza e sulla concorrenza. È stato sviluppato da Mozilla Research con l'obiettivo di essere un linguaggio di programmazione moderno, concorrente e sicuro. 35

**Smart Home** traducibile come *casa intelligente*, si riferisce a un sistema di automazione domestica che fornisce agli utenti il controllo remoto di elettrodomestici e dispositivi domestici come luci, porte, termostati, videocamere di sicurezza, allarmi antincendio e altro ancora. 6

**Timestamp** traducibile come *marca temporale*, è una sequenza di caratteri o di codice generata da un algoritmo di hashing che identifica in maniera univoca un determinato blocco di dati. 28, 29

**TPM** traducibile come *modulo di piattaforma fidata*, è un chip integrato in un computer che fornisce funzionalità di crittografia. Il suo scopo principale è quello di generare, archiviare e limitare l'accesso alle chiavi crittografiche. 45

**Zero-Day Vulnerability** si riferisce ad una vulnerabilità di sicurezza informatica che non è stata ancora scoperta o che non è stata ancora resa pubblica. Questo tipo di vulnerabilità consente agli attori malevoli di sfruttare i difetti del software o del firmware senza che il produttore ne sia a conoscenza. 21

# Bibliografia

## Articoli


[3] Ó. Gil, A. Garrell e A. Sanfeliu, «Social robot navigation tasks: Combining machine learning techniques and social force model,» *Sensors*, vol. 21, n. 21, p. 7087, 2021 (cit. alle pp. 4, 37).

[5] P. Foggia, A. Greco, A. Roberto, A. Saggese e M. Vento, «A social robot architecture for personalized real-time human-robot interaction,» *IEEE Internet of Things Journal*, 2023 (cit. alle pp. 4, 19).

[6] C. S. González-González, V. Violant-Holz e R. M. Gil-Iranzo, «Social robots in hospitals: a systematic review,» *Applied Sciences*, vol. 11, n. 13, p. 5976, 2021 (cit. alle pp. 4, 19).

[7] A. M. Aroyo, F. Rea, G. Sandini e A. Sciutti, «Trust and Social Engineering in Human Robot Interaction: Will a Robot Make You Disclose Sensitive Information, Conform to Its Recommendations or Gamble?» *IEEE Robotics and Automation Letters*, vol. 3, pp. 3701–3708, 4 ott. 2018, ISSN: 23773766. DOI: 10.1109/LRA.2018.2856272 (cit. alle pp. 4, 20).

[8] P. Pennisi, A. Tonacci, G. Tartarisco et al., «Autism and social robotics: A systematic review,» *Autism Research*, vol. 9, n. 2, pp. 165–183, 2016 (cit. alle pp. 4, 19).

[10] H. L. Cao, P. G. Esteban, A. D. Beir, G. V. D. Perre, R. Simut e B. Vanderborght, «A platform-independent robot control architecture for multiple therapeutic scenarios,» *AISB Annual Convention 2016, AISB 2016*, vol. 1, pp. 1–5, lug. 2016. indirizzo: https://arxiv.org/abs/1607.04971v1 (cit. alle pp. 4, 19).

[11] R. Subramanian, «Emergent AI, Social Robots and the Law: Security, Privacy and Policy Issues,» *Journal of International Technology and Information Management*, vol. 26, p. 2017, 2017. indirizzo: http://scholarworks.lib.csusb.edu/jitimAvailableat:http://scholarworks.lib.csusb.edu/jitim/vol26/iss3/4Electroniccopyavailableat:https://ssrn.com/abstract=3279236 (cit. alle pp. 4, 20).

[13] M. Malfaz, Á. Castro-González, R. Barber e M. A. Salichs, «A biologically inspired architecture for an autonomous and social robot,» *IEEE Transactions on Autonomous Mental Development*, vol. 3, n. 3, pp. 232–246, 2011 (cit. alle pp. 4, 19).

[14] N. Lazzeri, D. Mazzei, L. Cominelli, A. Cisternino e D. E. De Rossi, «Designing the mind of a social robot,» *Applied Sciences*, vol. 8, n. 2, p. 302, 2018 (cit. alle pp. 4, 19).

[16] S. Keizer, M. Ellen Foster, Z. Wang e O. Lemon, «Machine learning for social multiparty human–robot interaction,» *ACM transactions on interactive intelligent systems (TIIS)*, vol. 4, n. 3, pp. 1–32, 2014 (cit. a p. 4).

[17] S. O. Oruma, M. Sánchez-Gordón, R. Colomo-Palacios, V. Gkioulos e J. K. Hansen, «A Systematic Review on Social Robots in Public Spaces: Threat Landscape and Attack Surface,» *Computers*, vol. 11, pp. 1–45, 12 dic. 2022, ISSN: 2073431X. DOI: 10.3390/COMPUTERS11120181 (cit. alle pp. 5, 19).

[30] C. Dwork, F. McSherry, K. Nissim e A. Smith, «Calibrating Noise to Sensitivity in Private Data Analysis,» *Journal of Privacy and Confidentiality*, vol. 7, pp. 17–51, 3 mag. 2017, ISSN: 2575-8527. DOI: 10.29012/jpc.v7i3.405. indirizzo: https://journalprivacyconfidentiality.org/index.php/jpc/article/view/405 (cit. a p. 9).

[31] Z. Ding, Y. Wang, G. Wang, D. Zhang e D. Kifer, «Detecting Violations of Differential Privacy,» vol. 15, 2018. DOI: 10.1145/3243734.3243818. indirizzo: https://doi.org/10.1145/3243734.3243818 (cit. a p. 9).

[32] M. Bi, Y. Wang, Z. Cai e X. Tong, «A privacy-preserving mechanism based on local differential privacy in edge computing,» *China Communications*, vol. 17, pp. 50–65, 9 set. 2020, ISSN: 16735447. DOI: 10.23919/JCC.2020.09.005 (cit. a p. 9).

[33] F. Faisal, N. Mohammed, C. K. Leung e Y. Wang, «Generating Privacy Preserving Synthetic Medical Data,» *Proceedings - 2022 IEEE 9th International Conference on Data Science and Advanced Analytics, DSAA 2022*, 2022. DOI: 10.1109/DSAA54385.2022.10032429 (cit. a p. 9).

[34] T. Komarova e D. Nekipelov, «Identification and Formal Privacy Guarantees,» *SSRN Electronic Journal*, giu. 2020. DOI: 10.2139/ssrn.3635824. indirizzo: https://arxiv.org/abs/2006.14732v2 (cit. a p. 9).

[48] L. Ragno, A. Borboni, F. Vannetti, C. Amici e N. Cusano, «Application of Social Robots in Healthcare: Review on Characteristics, Requirements, Technical Solutions,» *Sensors*, vol. 23, n. 15, p. 6820, 2023 (cit. a p. 19).

[50] S. Coşar, M. Fernandez-Carmona, R. Agrigoroaie et al., «ENRICHME: Perception and Interaction of an Assistive Robot for the Elderly at Home,» *International Journal of Social Robotics*, vol. 12, pp. 779–805, 2020 (cit. a p. 19).

[51] N. Hendrich, H. Bistry e J. Zhang, «Architecture and software design for a service robot in an elderly-care scenario,» *Engineering*, vol. 1, n. 1, pp. 027–035, 2015 (cit. a p. 19).

[52] L. Asprino, P. Ciancarini, A. G. Nuzzolese, V. Presutti e A. Russo, «A reference architecture for social robots,» *Journal of Web Semantics*, vol. 72, p. 100 683, apr. 2022, ISSN: 1570-8268. DOI: 10.1016/J.WEBSEM.2021.100683 (cit. a p. 19).

[53] P. Alves-Oliveira, S. Gomes, A. Chandak, P. Arriaga, G. Hoffman e A. Paiva, «Software architecture for YOLO, a creativity-stimulating robot,» *SoftwareX*, vol. 11, p. 100 461, 2020 (cit. a p. 19).

[54] H.-L. Cao, G. Van de Perre, J. Kennedy et al., «A personalized and platform-independent behavior control system for social robots in therapy: development and applications,» *IEEE Transactions on Cognitive and Developmental Systems*, vol. 11, n. 3, pp. 334–346, 2018 (cit. a p. 19).

[55] E. Coronado, D. Deuff, P. Carreno-Medrano et al., «Towards a modular and distributed end-user development framework for human-robot interaction,» *IEEE Access*, vol. 9, pp. 12 675–12 692, 2021 (cit. a p. 19).

[56] B. Graf, M. Hans e R. D. Schraft, «Care-O-bot II–Development of a next generation robotic home assistant,» *Autonomous robots*, vol. 16, n. 2, pp. 193–205, 2004 (cit. a p. 19).

[57] M. A. Salichs, Á. Castro-González, E. Salichs et al., «Mini: a new social robot for the elderly,» *International Journal of Social Robotics*, vol. 12, pp. 1231–1249, 2020 (cit. a p. 19).

[58] M. Bonaccorsi, L. Fiorini, F. Cavallo, A. Saffiotti e P. Dario, «A cloud robotics solution to improve social assistive robots for active and healthy aging,» *International Journal of Social Robotics*, vol. 8, pp. 393–408, 2016 (cit. alle pp. 19, 39, 40).

[59] D. Loza-Matovelle, A. Verdugo, E. Zalama e J. Gómez-García-Bermejo, «An architecture for the integration of robots and sensors for the care of the elderly in an ambient assisted living environment,» *Robotics*, vol. 8, n. 3, p. 76, 2019 (cit. alle pp. 19, 39, 40).

[60] S. Chatterjee, R. Chaudhuri e D. Vrontis, «Usage Intention of Social Robots for Domestic Purpose: From Security, Privacy, and Legal Perspectives,» *Information Systems Frontiers*, 2021, ISSN: 15729419. DOI: 10.1007/s10796-021-10197-7 (cit. a p. 19).

[63] K. Abouelmehdi, A. Beni-Hessane e H. Khaloufi, «Big healthcare data: preserving security and privacy,» *Journal of big data*, vol. 5, n. 1, pp. 1–18, 2018 (cit. a p. 20).

[64] H. Huang, P. Zhu, F. Xiao, X. Sun e Q. Huang, «A blockchain-based scheme for privacy-preserving and secure sharing of medical data,» *Computers & Security*, vol. 99, p. 102 010, dic. 2020, ISSN: 0167-4048. DOI: 10.1016/J.COSE.2020.102010 (cit. alle pp. 20, 39).

[65] A. E. B. Tomaz, J. C. D. Nascimento, A. S. Hafid e J. N. D. Souza, «Preserving privacy in mobile health systems using non-interactive zero-knowledge proof and blockchain,» *IEEE Access*, vol. 8, pp. 204 441–204 458, 2020, ISSN: 21693536. DOI: 10.1109/ACCESS.2020.3036811 (cit. alle pp. 20, 39, 40).

[66] C. Lutz, M. Schöttler e C. P. Hoffmann, «The privacy implications of social robots: Scoping review and expert interviews,» *Mobile Media & Communication*, vol. 7, n. 3, pp. 412–434, 2019 (cit. a p. 20).

[67] T. Heuer, I. Schiering e R. Gerndt, «Privacy-centered design for social robots,» *Interaction Studies*, vol. 20, n. 3, pp. 509–529, 2019 (cit. a p. 20).

[68] E. Parliament, «REGOLAMENTO (UE) 2016/679 DEL PARLAMENTO EUROPEO E DEL CONSIGLIO del 27 aprile 2016 relativo alla protezione delle persone fisiche con riguardo al trattamento dei dati personali, nonchè alla libera circolazione di tali dati e che abroga la direttiva 95/46/CE (regolamento generale sulla protezione dei dati),» 2016 (cit. alle pp. 20, 22).

[69] E. Parliament, «Blockchain and the General Data Protection Regulation Can distributed ledgers be squared with European data protection law?,» 2016. DOI: 10.2861/535. indirizzo: http://www.europarl.europa.eu/thinktank (cit. a p. 20).

[70] U. Pagallo, «Robots in the cloud with privacy: A new threat to data protection?» *Computer Law & Security Review*, vol. 29, n. 5, pp. 501–508, 2013 (cit. a p. 20).

[71] A. Wright, S. Aaron e D. W. Bates, «The big phish: cyberattacks against US healthcare systems,» *Journal of General Internal Medicine*, vol. 31, pp. 1115–1118, 2016 (cit. a p. 21).

[73] H. Pirayesh e H. Zeng, «Jamming attacks and anti-jamming strategies in wireless networks: A comprehensive survey,» *IEEE communications surveys & tutorials*, vol. 24, n. 2, pp. 767–809, 2022 (cit. a p. 21).

[76] K. McKay e D. Cooper, «Guidelines for the selection, configuration, and use of transport layer security (TLS) implementations,» 2017 (cit. alle pp. 21, 26).

[77] K. Zhang, K. Yang, X. Liang, Z. Su, X. Shen e H. H. Luo, «Security and privacy for mobile healthcare networks: from a quality of protection perspective,» *IEEE Wireless Communications*, vol. 22, n. 4, pp. 104–112, 2015 (cit. alle pp. 22, 24).

[78] S. K. Kharroub, K. Abualsaud e M. Guizani, «Medical IoT: A comprehensive survey of different encryption and security techniques,» *2020 International Wireless Communications and Mobile Computing (IWCMC)*, pp. 1891–1896, 2020 (cit. alle pp. 22, 24).

[79] S. S. Bhuyan, H. Kim, O. O. Isehunwa et al., «Privacy and security issues in mobile health: Current research and future directions,» *Health policy and technology*, vol. 6, n. 2, pp. 188–191, 2017 (cit. alle pp. 22, 24).

[80] M. M. de Graaf, S. Ben Allouch e J. A. Van Dijk, «Why would I use this in my home? A model of domestic social robot acceptance,» *Human–Computer Interaction*, vol. 34, n. 2, pp. 115–173, 2019 (cit. a p. 22).

[81] A. Alrawais, A. Alhothaily, X. Cheng, C. Hu e J. Yu, «SecureGuard: A Certificate Validation System in Public Key Infrastructure,» *IEEE Transactions on Vehicular Technology*, vol. 67, n. 6, pp. 5399–5408, giu. 2018, ISSN: 1939-9359. DOI: 10.1109/TVT.2018.2805700 (cit. a p. 24).

[83] J. Bethencourt, A. Sahai e B. Waters, «Ciphertext-policy attribute-based encryption,» *Proceedings - IEEE Symposium on Security and Privacy*, pp. 321–334, 2007, ISSN: 10816011. DOI: 10.1109/SP.2007.11 (cit. alle pp. 25, 35, 36).

[84] F. B. Schneider, «Least privilege and more [computer security],» *IEEE Security & Privacy*, vol. 1, n. 5, pp. 55–59, 2003 (cit. a p. 27).

[85] R. C. Merkle, «Secure Communications over Insecure Channels,» *Commun. ACM*, vol. 21, n. 4, pp. 294–299, apr. 1978, ISSN: 0001-0782. DOI: 10.1145/

359460.359473. indirizzo: https://doi.org/10.1145/359460.359473 (cit. alle pp. 27, 28).

[89] H. Aamot, C. D. Kohl, D. Richter e P. Knaup-Gregori, «Pseudonymization of patient identifiers for translational research,» *BMC Medical Informatics and Decision Making*, vol. 13, pp. 1–15, 1 lug. 2013, ISSN: 14726947. DOI: 10.1186/1472-6947-13-75/FIGURES/8. indirizzo: https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/1472-6947-13-75.

[90] A. Ahmad e M. A. Babar, «Software architectures for robotic systems: A systematic mapping study,» *Journal of Systems and Software*, vol. 122, pp. 16–39, dic. 2016, ISSN: 0164-1212. DOI: 10.1016/J.JSS.2016.08.039.

[91] P. Asgharian, A. M. Panchea e F. Ferland, «A review on the use of mobile service robots in elderly care,» *Robotics*, vol. 11, n. 6, p. 127, 2022.

[92] S. Berloto, E. Notarnicola, E. Perobelli e A. Rotolo, «Italy and the COVID-19 long-term care situation,» *International Long Term Care Policy Network*, 2020.

[93] P. M. Chanal e M. S. Kakkasageri, «Security and Privacy in IoT: A Survey,» *Wireless Personal Communications*, vol. 115, pp. 1667–1693, 2 nov. 2020, ISSN: 1572834X. DOI: 10.1007/S11277-020-07649-9/FIGURES/8. indirizzo: https://link.springer.com/article/10.1007/s11277-020-07649-9.

[94] A. Channa, N. Popescu, J. Skibinska e R. Burget, «The rise of wearable devices during the COVID-19 pandemic: A systematic review,» *Sensors*, vol. 21, n. 17, p. 5787, 2021.

[95] A. Dillon, «User acceptance of information technology,» *Encyclopedia of human factors and ergonomics*, vol. 1, pp. 1105–1109, 2001.

[97] H. Guo e X. Yu, «A survey on blockchain technology and its security,» *Blockchain: Research and Applications*, vol. 3, p. 100 067, 2 giu. 2022, ISSN: 2096-7209. DOI: 10.1016/J.BCRA.2022.100067.

[98] M. A. J. Jamali, A. Heidari, P. Allahverdizadeh, F. Norouzi e B. Bahrami, «IoT Architecture,» *EAI/Springer Innovations in Communication and Computing*, pp. 9–31, 2020, ISSN: 25228609. DOI: 10.1007/978-3-030-18468-1_2/FIGURES/12. indirizzo: https://link.springer.com/chapter/10.1007/978-3-030-18468-1%5C%5F2.

[99] A. S. Khan, K. Balan, Y. Javed, J. Abdullah e S. Tarmizi, «Secure Trust-Based Blockchain Architecture to Prevent Attacks in VANET,» *Sensors 2019, Vol. 19, Page 4954*, vol. 19, p. 4954, 22 nov. 2019, ISSN: 1424-8220. DOI: 10.3390/S19224954. indirizzo: https://www.mdpi.com/1424-8220/19/22/4954/htm%20https://www.mdpi.com/1424-8220/19/22/4954.

[100] P. Kodeswaran e E. Viegas, «A policy based infrastructure for social data access with privacy guarantees,» *Proceedings - 2010 IEEE International Symposium on Policies for Distributed Systems and Networks, Policy 2010*, pp. 14–17, 2010. DOI: 10.1109/POLICY.2010.25.

[101] Cergas Bocconi, «Care for the elderly in Italy: preventing non-self-sufficiency from a young age and shortage of nurses among the main data,» *4th Long-Term Care Observatory Report*, 2022. indirizzo: https://rb.gy/649p0a.

[102] S. Nakamoto, «Bitcoin: A Peer-to-Peer Electronic Cash System,» pp. 1–9, 2008. indirizzo: www.bitcoin.org.

[104] T. Neubauer e J. Heurix, «A methodology for the pseudonymization of medical data,» *International journal of medical informatics*, vol. 80, n. 3, pp. 190–204, 2011.

[105] R. Pise e S. Patil, «A Deep Dive into Blockchain-based Smart Contract-specific Security Vulnerabilities,» *2022 IEEE International Conference on Blockchain and Distributed Systems Security, ICBDS 2022*, 2022. DOI: 10.1109/ICBDS53701.2022.9935949.

[106] L. Pycroft e T. Z. Aziz, «Security of implantable medical devices with wireless connections: The dangers of cyber-attacks,» *Expert Review of Medical Devices*, vol. 15, n. 6, pp. 403–406, 2018.

[108] S. U. R. Aqeel-ur-Rehman, I. U. Khan, M. Moiz e S. Hasan, «Security and privacy issues in IoT,» *International Journal of Communication Networks and Information Security (IJCNIS)*, vol. 8, n. 3, pp. 147–157, x 2016. DOI: 10.17762/ijcnis.v8i3.2074. indirizzo: https://www.researchgate.net/profile/Malaika-Moiz-2/publication/313574376%5C%5FSecurity%5C%5Fand%5C%5Fprivacy%5C%5Fissues%5C%5Fin%5C%5FIoT/links/5f8e1613458515b7cf8dbd62/Security-and-Privacy-Issues-in-IoT.pdf.

[109] S. Sarawi, M. Anbar, K. Alieyan, M. S. Alzubaidi, S. Al-Sarawi e M. Alzubaidi, «Internet of Things (IoT) communication protocols,» *ieeexplore.ieee.org*, 2017, Documento utile per capire le funzionalità di base dei protocolli. DOI: 10.1109/ICITECH.2017.8079928. indirizzo: https://ieeexplore.ieee.org/abstract/document/8079928/.

[110] Shruti, S. Rani, D. K. Sah e G. Gianini, «Attribute-Based Encryption Schemes for Next Generation Wireless IoT Networks: A Comprehensive Survey,» *Sensors 2023, Vol. 23, Page 5921*, vol. 23, p. 5921, 13 giu. 2023, ISSN: 1424-8220. DOI: 10.3390/S23135921. indirizzo: https://www.mdpi.com/1424-8220/23/13/5921/htm%20https://www.mdpi.com/1424-8220/23/13/5921.

[112] Z. Vahdati, A. Ghasempour, M. Salehi, S. M. Yasin e S. Yasin, «COMPARISON OF ECC AND RSA ALGORITHMS IN IOT DEVICES,» *Article in Journal of Theoretical and Applied Information Technology*, vol. 31, p. 16, 2019, ISSN: 1817-3195. indirizzo: https://www.researchgate.net/publication/335540942.

[113] I. Volkov e G. Radchenko, «Architecture of mHealth Platform for Storing, Exchanging and Processing of Medical Data in Smart Healthcare,» *Proceedings - 2021 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology, USBEREIT 2021*, pp. 117–120, mag. 2021. DOI: 10.1109/USBEREIT51232.2021.9455081.

[114] A. Vulpe, A. Paikan, R. Craciunescu et al., «IoT Security Approaches in Social Robots for Ambient Assisted Living Scenarios,» *International Symposium on Wireless Personal Multimedia Communications, WPMC*, vol. 2019-November, nov. 2019, ISSN: 13476890. DOI: 10.1109/WPMC48795.2019.9096127.

[115] G. Wang, R. Lu e Y. L. Guan, «Achieve Privacy-Preserving Priority Classification on Patient Health Data in Remote eHealthcare System,» *IEEE Access*, vol. 7, pp. 33 565–33 576, 2019, ISSN: 21693536. DOI: 10.1109/ACCESS.2019.2891775.

[116] M. Wazid, A. K. Das, N. Kumar et al., «Design of Lightweight Authentication and Key Agreement Protocol for Vehicular Ad Hoc Networks,» *IEEE Access*, vol. 5, pp. 14 966–14 980, lug. 2017, ISSN: 21693536. DOI: 10.1109/ACCESS.2017.2723265.

[117] Z. Wenhua, F. Qamar, T. A. N. Abdali, R. Hassan, S. T. A. Jafri e Q. N. Nguyen, «Blockchain Technology: Security Issues, Healthcare Applications, Challenges and Future Trends,» *Electronics (Switzerland)*, vol. 12, 3 2023, ISSN: 20799292. DOI: 10.3390/electronics12030546.

## Atti

[4] S. Jeong, D. E. Logan, M. S. Goodwin et al., «A social robot to mitigate stress, anxiety, and pain in hospital pediatric care,» in *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction Extended Abstracts*, 2015, pp. 103–104 (cit. alle pp. 4, 19).

[9] T. Kanda, M. Shimada e S. Koizumi, «Children learning with a social robot,» in *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, 2012, pp. 351–358 (cit. alle pp. 4, 19).

[12] P. Salvini, G. Ciaravella, W. Yu et al., «How safe are service robots in urban environments? Bullying a robot,» in *19th international symposium in robot and human interactive communication*, IEEE, 2010, pp. 1–7 (cit. alle pp. 4, 40).

[15] P. Su e X. Yuan, «Are You Watching Me? A Study on Privacy Notice Design of Social Robot,» in *Advances in Ergonomics in Design: Proceedings of the AHFE 2021 Virtual Conference on Ergonomics in Design, July 25-29, 2021, USA*, Springer, 2021, pp. 339–344 (cit. alle pp. 4, 20).

[29] C. Dwork, «Differential privacy,» in *International colloquium on automata, languages, and programming*, Springer, 2006, pp. 1–12 (cit. a p. 9).

[49] S. Shin, D. Kang e S. S. Kwak, «Telepresence Robot for Isolated Patients in the COVID-19 Pandemic: Effects of Socio-relationship and Telecommunication Device Types on Patients' Acceptance of Robots,» in *International Conference on Social Robotics*, Springer, 2022, pp. 263–276 (cit. a p. 19).

[61] J. Miller, A. B. Williams e D. Perouli, «A case study on the cybersecurity of social robots,» in *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, 2018, pp. 195–196 (cit. alle pp. 19, 21).

[62] T. Denning, C. Matuszek, K. Koscher, J. R. Smith e T. Kohno, «A spotlight on security and privacy risks with future household robots: attacks and lessons,» in *Proceedings of the 11th international conference on Ubiquitous computing*, 2009, pp. 105–114 (cit. a p. 19).

[72] K. Man, Z. Qian, Z. Wang, X. Zheng, Y. Huang e H. Duan, «Dns cache poisoning attack reloaded: Revolutions with side channels,» in *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security*, 2020, pp. 1337–1350 (cit. a p. 21).

[87] S. Agrawal e M. Chase, «FAME: Fast Attribute-Based Message Encryption,» in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communi-*

*cations Security*, ser. CCS '17, Dallas, Texas, USA: Association for Computing Machinery, 2017, pp. 665–682, ISBN: 9781450349468. DOI: 10.1145/3133956.3134014. indirizzo: https://doi.org/10.1145/3133956.3134014 (cit. alle pp. 35, 36).

[96] C. Dwork, «Differential privacy: A survey of results,» in *International conference on theory and applications of models of computation*, Springer, 2008, pp. 1–19.

[103] H. T. Neprash, C. C. McGlave, D. A. Cross et al., «Trends in ransomware attacks on US hospitals, clinics, and other health care delivery organizations, 2016-2021,» in *JAMA Health Forum*, American Medical Association, vol. 3, 2022, e224873–e224873.

[107] D. Quarta, M. Pogliani, M. Polino, F. Maggi, A. M. Zanchettin e S. Zanero, «An Experimental Security Analysis of an Industrial Robot Controller,» Institute of Electrical e Electronics Engineers Inc., giu. 2017, pp. 268–285, ISBN: 9781509055326. DOI: 10.1109/SP.2017.20.

[111] K. Tanaka, R. Mayuzumi, T. Takahashi, S. Takaki e N. Oka, «Robot Mediated Handholding Combined with a Mobile Video Call Makes the Users Feel Nearer and Closer,» in *Proceedings of the 9th International Conference on Human-Agent Interaction*, 2021, pp. 3–12.

## Sitografia

[1] Spritz, *Spritz*. indirizzo: https://spritz.math.unipd.it/ (cit. a p. 2).

[2] Omitech, *Omitech*. indirizzo: https://www.omitech.it/ (cit. a p. 2).

[18] Sanbot, *Sanbot Elf*. indirizzo: http://en.sanbot.com/product/sanbot-elf/design (cit. a p. 5).

[19] Softbank, *Pepper Robot*. indirizzo: https://www.softbankrobotics.com/emea/en/pepper (cit. a p. 5).

[20] Amazon, *Amazon Echo Dot 4*. indirizzo: https://www.amazon.it/ (cit. a p. 7).

[21] Google, *Google Nest Hub*. indirizzo: https://store.google.com/it/product/nest_hub_2nd_gen?hl=it&pli=1 (cit. a p. 7).

[22] Samsung, *Samsung Smart Lock*. indirizzo: https://www.samsung.com/it/smartthings/do-the-smartthings/ (cit. a p. 7).

[23] Samsung, *Samsung Family Hub*. indirizzo: https://www.samsung.com/it/refrigerators/all-refrigerators/?family-hub (cit. a p. 7).

[24] Amazon, *Apple Watch Series 8*. indirizzo: https://www.amazon.it/ (cit. a p. 8).

[25] Samsung, *Samsung Galaxy Watch 4*. indirizzo: https://www.samsung.com/it/watches/?product1=sm-r930nzeaitv&product2=sm-r960nzkaitv&product3=sm-r920nztaitv (cit. a p. 8).

[26] Xiaomi, *Xiaomi Smart Band 6*. indirizzo: https://www.mi.com/global/mi-smart-band-6/ (cit. a p. 8).

[27] Abbott, *FreeStyle Libre*. indirizzo: https://www.freestyle.abbott/it-it/products/freestyle-libre-2.html (cit. a p. 8).

[28] ResearchGate, *Probabilistic Noise*. indirizzo: https://www.researchgate.net/figure/Examples-of-images-modified-by-Gaussian-noise-Gaussian-noise-was-applied-on-each-image_fig7_221913964 (cit. a p. 9).

[35] D. D. C. .-. Quora, *Public Key Cryptography*. indirizzo: https://www.quora.com/How-many-keys-are-used-in-public-key-cryptography (cit. a p. 10).

[36] MDPI, *A Comprehensive Survey on Security and Privacy for Electronic Health Data*. indirizzo: https://www.mdpi.com/1660-4601/18/18/9668 (cit. a p. 11).

[37] Springer, *SHARE-ABE*. indirizzo: https://link.springer.com/article/10.1007/s10586-021-03382-5/figures/4 (cit. a p. 11).

[38] Jgraph, *Draw.io*. indirizzo: https://app.diagrams.net/ (cit. a p. 15).

[39] Google, *Google Drive*. indirizzo: https://www.google.com/drive/ (cit. a p. 15).

[40] Google, *Google Docs*. indirizzo: https://www.google.com/docs/about/ (cit. a p. 15).

[41] Google, *Google Scholar*. indirizzo: https://scholar.google.com/ (cit. a p. 16).

[42] Google, *Google Sheets*. indirizzo: https://www.google.com/sheets/about/ (cit. a p. 16).

[43] Google, *Google Slides*. indirizzo: https://www.google.com/slides/about/ (cit. a p. 17).

[44] Mendeley, *Mendeley*. indirizzo: https://www.mendeley.com/ (cit. a p. 17).

[45] Oracle, *Virtualbox*. indirizzo: https://www.virtualbox.org/ (cit. a p. 17).

[46] Overleaf, *Overleaf*. indirizzo: https://www.overleaf.com/ (cit. a p. 18).

[47] Microsoft, *Visual Studio Code*. indirizzo: https://code.visualstudio.com/ (cit. a p. 18).

[74] Codefinity, *MITM*. indirizzo: https://codefinity.com/blog/Man%20in%20the%20middle%20attack (cit. a p. 22).

[75] Bluecat, *DNS Poisoning*. indirizzo: https://bluecatnetworks.com/blog/what-is-dns-poisoning-how-to-prevent-it/ (cit. a p. 23).

[82] O. W. J. et Al., *Certificate and Public Key Pinning*, https://owasp.org/www-community/controls/Certificate_and_Public_Key_Pinning# (cit. a p. 25).

[86] 2. Fraunhofer-AISEC, *RABE: Rust Attribute-Based Encryption*, https://github.com/Fraunhofer-AISEC/rabe, 2022 (cit. a p. 35).

[88] 2. dwyl.io - Do What You Love, *English Words*, https://github.com/dwyl/english-words/blob/master/words.txt, 2020 (cit. a p. 36).